Qute: Quality-of-Monitoring Aware Sensing and Routing Strategy in Wireless Sensor Networks

Shaojie Tang
Department of Computer and Information Sciences
Temple University
Philadelphia PA 19122
shaojie.tang@temple.edu

Jie Wu
Department of Computer and Information Sciences
Temple University
Philadelphia PA 19122
jiewu@temple.edu

ABSTRACT

Wireless Sensor Networks (WSNs) are widely used to monitor the physical environment. In a highly redundant sensor network, sensor readings from nearby sensors often have high similarity. In this work, we are interested in how to decide an appropriate sensing rate [1] for each sensor node, in order to maximize the overall Quality-of-Monitoring (QoM), while ensuring that all readings can be transmitted to the sink. Note that a feasible sensing rate allocation should satisfy both energy constraint on each sensor node and flow conservation through the network. In order to capture the statistical correlations among sensor readings, we first introduce the concept of correlation graph. The correlation graph is further decomposed into several *correlation components*, and sensor readings from the same correlation component are highly correlated. For each correlation component, we defined a general utility function to estimate the QoM. The utility function of each correlation component is a non-decreasing submodular function of the total sensing rates allocated to that correlation component. Then we formulate the QoM-aware sensing rate allocation problem as a utility maximization problem under limited power supply on each node. To tackle this problem, we adopted an efficient algorithm, called *Qute*, by jointly considering both the energy constraint on each node and flow conservation through the network. Under some settings, we analytically show that *Qute* can find the optimal QoM-aware sensing rate allocation which achieves the maximum total utility. We conducted extensive testbed verifications of our schemes, and experimental results validate our theoretical results.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication, Network topology; G.2.2 [**Graph Theory**]: Network problems, Graph algorithms

[1]The number of sensor readings per unit time.

MobiHoc'13, July 29–August 1, 2013, Bangalore, India.

Keywords

Sensing rate allocation, routing design, Quality-of-Monitoring.

1. INTRODUCTION

Wireless Sensor Networks (WSNs) often contain a large amount of sensor nodes which are spatially distributed to monitor physical or environmental conditions, such as temperature, humidity, *etc.*, over a geographic region. Different from traditional networking systems, developing an effective sensor network must take into account both its Quality-of-Monitoring (QoM) [1] and limited energy resource [2]. Sensor nodes periodically gather sensing values from its nearby environment, therefore the QoM crucially depends on the sensing rate of each sensor node. Typically, a high sensing rate is required to achieve high QoM. We note that most sensor nodes used in large scale sensing applications are often resource constrained, with an extremely limited energy budget and wireless communication ability. The massive amount of sensing data posed great challenges on designing efficient data gathering schemes under various network resource constraints. Therefore, there is a great need for developing a QoM-aware sensing rate allocation scheme to decide an appropriate sensing rate for each node, in order to maximize the overall QoM subject to energy constraint on each node. One naive method is to treat each sensor node equally and collect as many readings as possible from the entire network. However, this approach ignores the underlying correlation among sensor nodes. We observed that the sensor readings from nearby sensors are often correlated, resulting in inter-node dependency [3] [4] [5] [6] [7] [8]; This provides us with a good opportunity to design a better sensing rate allocation scheme by avoiding redundant sensor readings.

In this work, we first introduce the correlation graph [4] [9] to capture the statistical correlations among sensor nodes. We further partition the correlation graph into several *correlation components* [10] [11] such that sensor readings from the same component are *highly* correlated. In particular, given the sensor readings gathered from some individual nodes from one correlation component, we can use interpolation to estimate the readings at all other nodes in that component. For each correlation component, we define a general utility function to quantify the Quality-of-Monitoring. The utility function is a non-decreasing submodular function, depending on the total sensing rates from all sensors within that correlation component. The submodularity of the QoM function is due to the correlations among sensor nodes within one component. For example, when using WSNs to monitor the humidity in the forest, we observe that the readings from nearby sensors are often correlated with each other. It implies that allocating an additional sampling to one component results in diminishing improvement to the QoM, as the amount of samplings allocated to the same component grows. The submodularity of QoM for monitoring physical phenomena has been observed in many real-world data sets, including target tracking [12], water quality monitoring [13] and temperature monitoring [14]. In this work, we aim to exploit such correlation to develop a QoM-aware sensing rate allocation scheme, in order to maximize the overall utility, summed over all correlation components. The main contributions of this paper are as follows.

(1) We introduce the concept of correlation component and propose a simple yet general representation, called *utility function* of QoM, which captures the data correlation relationship among sensor nodes in the same correlation component. Particularly, we reveal space-dependence by characterizing the utility function for each correlation component as a non-decreasing submodular function of the total sensing rate from that component.

(2) Based on the proposed utility function, we jointly consider sensing rate allocation and routing design under the energy constraint on each node, and pose those two techniques into a uniform framework, called *Qute*. We theoretically prove that *Qute* can achieve the maximum total utility if the energy consumption for sensing unit data is no less than that for receiving unit data.

(3) We conducted extensive testbed verifications of our protocols, and experimental results show that our protocols perform well in practice, in terms of utility maximization and estimation error minimization.

The remainder of the paper is organized as follows: Section 2 summarizes the related works; Section 3 discusses the motivation of this work and introduces the problem formulation; we adopt an efficient rate allocation protocol in Section 4; some interesting extensions of this work are discussed in Section 5; extensive experimental results are reported in Section 6; we conclude this paper in Section 7.

2. RELATED WORK

Exploiting the correlations among sensor readings in large scale WSNs attracts increasing interest in [1] [5] [15] [16] [17] [18]. In [7] [19], they develop an efficient communication protocol by taking advantage of sensor reading correlations. Our work differs from theirs in the sense that we put our focus on designing a joint sensing rate allocation and routing design, by exploiting spatial correlations among nodes.

In [20] [21], lexicographic maxmin fairness for data collection in traditional WSN and solar powered WSN has been extensively studied. Their objective is to maximize the minimum sensing rate among all sensors, while adhering to energy constraints on each sensor node. In this work, we show that our problem is closely related to fair rate allocation problem, therefore it allows us to adopt a similar rate allocation scheme.

Another category of related works studies the submodularity of sensor readings in various WSN applications such as temperature monitoring [14], water quality monitoring [13], and target tracking [12]. A QoM-aware sensing scheduling scheme is developed in [2]. They propose a simple greedy algorithm for deciding the sensing activity of different sensors. Previous work is extended in [8], they characterize the QoM from both time and space domain as a submodular function. They propose several sensing scheduling schemes under both the centralized and distributed settings. Very recently, Tang *et al.* [22] revisited the distributed sensing scheduling problem, they formulated the problem as a potential game by treating each sensor node as a player, and the convergence of their scheme is guaranteed.

3. MOTIVATIONS AND PROBLEM FORMULATION

3.1 Motivating Application

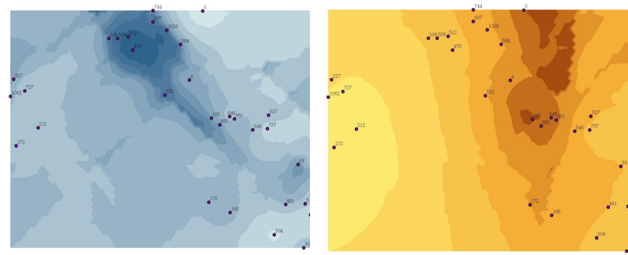

Figure 1: Humidity (left) and temperature (right) distribution over a forest. Dark dots represent the locations of different sensors that are deployed in the forest.

This work was originally motivated by Greenorbs [23] in which thousands of sensors are deployed in the forest for environment monitoring. Each sensor node periodically measures several signal values (such as temperature, humidity, illuminance, and CO_2) from the environment, and continuously transmits them back to the base station through multi-hop relay. To achieve satisfactory quality of monitoring, typical WSN applications require spatially dense sensor deployment [6]. As a result, the readings among neighboring nodes are often spatially correlated. The degree of correlation depends on the internode separation [19]. This kind of spatial redundancy information is referred to as the 'spatial correlation'. For instance, Figure 1 plots the temperature and humidity distribution over a forest, with different colors indicating different senor readings from corresponding areas. We observe that both temperature and humidity readings from nearby sensor nodes tend to be similar to one another; the degree of similarity varies according their locations and spatial distribution of temperature. Those sensors with similar readings naturally form a component or cluster.

Our work aims at QoM-aware sensing rate allocation by exploiting correlation among sensor nodes. We treat all the sensor nodes in one component equally, and define a utility function to quantify the QoM for each component under various sensing rate allocations. The utility function is a submodular function related to total sensing rate from that component. We intend to find the sensing rate for each node, in order to maximize the overall utility, by minimizing the redundant sensor readings from the same component. Note that a feasible rate allocation should take into consideration both the energy budget of each node, and the flow conservation through the network.

3.2 System Models

We first introduce several preliminary concepts which will be frequently used throughout this paper.

3.2.1 Networking Model

Assume a wireless sensor network is composed of n sensor nodes, $\mathcal{V} = \{v_1, v_2, \cdots, v_n\}$, and each sensor has a fixed transmission range. The energy budget of sensor v per unit time is $\mathbf{B}_v$, and the energy consumption for transmitting unit data is δ_t, the energy consumption for receiving unit data is δ_r, the energy consumption for sensing (or generating) unit data is δ_s. We use s_v to represent the sensing rate (the number of generated samplings per unit time) of node v. $S = \{s_{v_1}, s_{v_2}, \cdots, s_{v_n}\}$ is a sensing rate allocation.

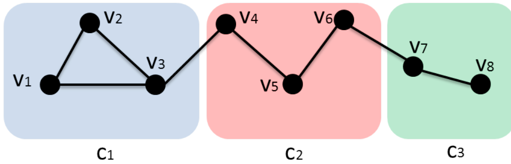

Figure 2: Communication graph and correlation components. Nodes v_1, v_2, v_3 belong to component $\mathbf{c}_1$; nodes v_4, v_5, v_6 belong the component $\mathbf{c}_2$; nodes v_7, v_8 belong to component $\mathbf{c}_3$.

Notice that after each sensor node generates the sensing data, it further transmits the sensing data to the sink node through single- or multi- hop routing. Therefore, a feasible sensing rate allocation should satisfy both energy constraint on each node, and flow conservation through the network.

DEFINITION 1 (COMMUNICATION GRAPH). *Given a sensor network consisting of a set of n sensors, the communication graph over the sensor network is a undirected graph with $\mathcal{V}$ as a set of vertices, and there exists an edge between any two sensors if, and only if, they can communicate with each other.*

3.2.2 Correlation Model

In this work, we use a correlation graph to represent the correlation among sensor nodes. Various graphical models, including Markov random files and Bayesian networks, are widely adopted to represent the statistical correlations among the sensors. Those models [3] [9] are also helpful for designing distributed estimation algorithms. Given a set of n sensors, $\mathcal{V} = \{v_1, v_2, \cdots, v_n\}$, a typical correlation graph of the sensor network is a undirected graph with $\mathcal{V}$ as a set of vertices, and there is a (maybe weighted) edge between any two sensors if, and only if, there exists a conditional dependency or partial correlation between them. The degree of the correlation or similarity between a pair of nodes can be estimated in many ways. One method treats the sensor reading of each node by a variable [3], then the correlation between two sensors is evaluated by the statistical dependence between those two variables. Another method [11] measures the difference between two time series, using the magnitude and the trend of time series. Details of the correlation graph construction is out of the scope of this paper.

According to the correlation graph built in the first phase, we partition all nodes into m disjoint components, $\mathcal{C} = \{\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_m\}$, called *correlation components* by adopting similar approaches used in [3] [4] [5] [10] [11] [24], such that the sensor readings reported by sensor nodes within the same component are highly correlated.

DEFINITION 2 (CORRELATION COMPONENT). *A correlation component is a subset of sensors where the sensor nodes within one component have similar sensing values. Thus, the sensing value reported by any sensor can be approximated or estimated by the readings of any sensor node within the same correlation component.*

Many efficient partitioning algorithms have been proposed to realize the partitioning, based on spatial correlation. In this work, we follow the same clustering techniques as in [11] to partition the correlation graph into a minimum number of cliques, and each clique is treated as a correlation component. As an example, Figure 2 illustrates a typical communication graph and correlation component. Sensor nodes within the same block belong to one correlation component. There is an edge between two nodes if they can communicate with each other.

3.2.3 Utility Function of QoM

Similar to [1] [8], we define a general submodular function to quantify the Quality-of-Monitoring (QoM) under different sensing rate allocations. Depending on the applications, different sensing rate allocations will provide different levels of QoM. Specifically, for a single component, the contribution of an additional sampling to the QoM crucially depends on the current sensing rate of that component. As pointed out in the literature [1] [8] [25], due to spatial-temporal correlation among sensor readings, the utility function of total sensing rate exhibits diminishing marginal returns. We were motivated by those observations to treat all the nodes in one component equally; we define the utility function, $\mathcal{U}^{\mathbf{c}_i}$, for each correlation component $\mathbf{c}_i$ by a general, non-decreasing, submodular function $\mathcal{U}()$ in terms of total sensing rate that is allocated to that component: $\mathcal{U}^{\mathbf{c}_i} = \mathcal{U}(\sum_{v\in\mathbf{c}_i} s_v)$. We say that $\mathcal{U}()$ is submodular if it satisfies a diminishing returns property: the difference from adding an element c to a set a is at least as large as the one from adding the same element to a superset b. The overall utility is defined by summing utilities over all correlation components:

$$\mathbf{U} = \sum_{\mathbf{c}_i\in\mathcal{C}} \mathcal{U}^{\mathbf{c}_i} = \sum_{\mathbf{c}_i\in\mathcal{C}} \mathcal{U}(\sum_{v\in\mathbf{c}_i} s_v)$$

3.3 Problem Formulation

Now we can present the formulation of the problem. Without loss of generality, we assume that there is a set of sensor nodes deployed over a two-dimensional area. In addition, there is one sink node to collect all sensing data from the network. The locations of the sensor nodes are fixed and known a priori, and the correlation components are predefined. We assume that each sensor performs a sensing task, *e.g.*, temperature sampling, at certain rate, and then transmits the sensed data to the sink node through single- or multi-hop routing.

In the remainder of this paper, we use f_{uv} to represent the amount of flow from sensor u to sensor v; s_u to represent the sensing rate of node u; $\mathbf{B}_u$ to denote the energy budget for node u; δ_t to represent the energy consumption for transmitting unit data; δ_r to represent the energy consumption for receiving unit data; δ_s to represent the energy consumption for sensing unit data. Then the QoM-aware rate allocation problem, given energy budget on each sensor node, is formulated as:

Problem: *QoM-aware Rate Allocation*
Objective: Maximize $\mathbf{U} = \sum_{\mathbf{c}_i\in\mathcal{C}} \mathcal{U}(\sum_{v\in\mathbf{c}_i} s_v)$
subject to:

$$\begin{cases} (1)\ \ s_u + \sum_{v\in N_u} f_{vu} = \sum_{v\in N_u} f_{uv}, \forall u \neq \text{ sink} \\ (2)\ \ \sum_{v\in N_u} f_{vu}\delta_r + \sum_{v\in N_u} f_{vu}\delta_t + s_u\delta_s \leq \mathbf{B}_u, \forall u \neq \text{ sink} \\ (3)\ \ f_{uv} \geq 0, \forall u, v \in \mathcal{V} \end{cases}$$

In the above formulation, N_v represents the set of v's neighbors in the communication graph. Constraint 1 specifies the flow conservation: total amount of inflow plus self-generated data is equal to that of outflow. Constraint 2 specifies that the total energy consumption on each node should not exceed its energy budget. The general objective of this work is to decide an appropriate sensing rate s_v for each node $v \in \mathcal{V}$ and associated flow assignment f_{uv} on each link uv in order to maximize the overall utility while satisfying both the energy constraint and flow conservation.

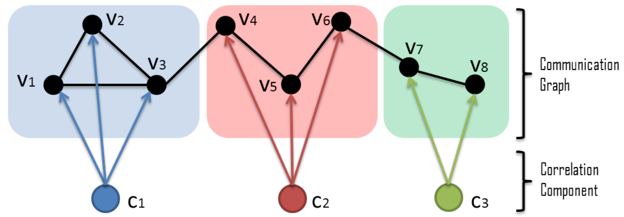

Figure 3: Two-layered Communication-Correlation Graph based on the example in Figure 2.

4. QOM-AWARE SENSING RATE ALLOCATION

We first introduce a two-layered Communication-Correlation Graph (CCG) by integrating the information from both communication graph and correlation component. Based on CCG, we introduce and study a new problem called *fair rate allocation* problem, which shares the same optimal solution with the original QoM-aware rate allocation problem in some scenarios. We adopt an efficient algorithm to find the optimal fair rate allocation, which is also the optimal QoM-aware rate allocation under some settings.

4.1 Communication-Correlation Graph

We first build the Communication-Correlation Graph (CCG) to integrate both the communication graph and the correlation graph. CCG provides a unified framework under which we are able to tackle the QoM-aware rate allocation problem by taking into consideration both the energy/flow constraint of underlying communication graph and the correlation relationship among sensor nodes. The construction of CCG, as a two-layered graph, takes as input the communication graph and the correlation components. The first layer is constituted by the communication graph, as defined in Section 3.2.1. In the second layer, we create a virtual node $\mathbf{c}_i$ for each correlation component, and we add a directed edge from virtual node $\mathbf{c}_i$ to sensor node v_i in the first layer if, and only if, $v_i \in \mathbf{c}_i$, as defined in Section 3.2.2. Please refer to Figure 3 to illustrate: this CCG is built based on the example shown in Figure 2, we add directed edges from $\mathbf{c}_1$ to v_1, v_2 and v_3; from $\mathbf{c}_2$ to v_4, v_5 and v_6; from $\mathbf{c}_3$ to v_7 and v_8.

Next, we re-formulate the QoM-aware rate allocation problem based on CCG. We assume there is no energy constraint on virtual node $\mathbf{c}_i$. The energy consumption for node v to receive unit data from adjacent $\mathbf{c}_i$ is δ_s, which is the sensing cost of node v in the original problem. Let $s_{\mathbf{c}}$ represent the total sensing rate from component $\mathbf{c}$: $s_{\mathbf{c}} = \sum_{v\in\mathbf{c}} s_v$. We use $S = \{s_{\mathbf{c}_1}, s_{\mathbf{c}_2}, \cdots, s_{\mathbf{c}_m}\}$ to represent a rate allocation where $s_{\mathbf{c}_i}$ is the sensing rate assigned to component $\mathbf{c}_i$. It was worth mentioning that in the new problem formulation, we define $s_{\mathbf{c}}$ to represent the total sensing rate of the entire component, instead of defining s_v for each individual node. Therefore, the overall utility function can be rewritten as:

$$\mathbf{U} = \sum_{\mathbf{c}\in\mathcal{C}} \mathcal{U}(\sum_{v\in\mathbf{c}} s_v) = \sum_{\mathbf{c}\in\mathcal{C}} \mathcal{U}(s_{\mathbf{c}})$$

Accordingly, we re-formulate the original problem as follows:

Problem: *QoM-Aware Rate Allocation based on CCG*
Objective: Maximize $\mathbf{U} = \sum_{\mathbf{c}\in\mathcal{C}} \mathcal{U}(s_{\mathbf{c}})$
subject to:

$$\begin{cases} (1)\ s_{\mathbf{c}} = \sum_{v\in N_c} f_{cv}, \forall \mathbf{c}\in\mathcal{C} \\ (2)\ \sum_{u\in N_v} f_{uv} + \underset{\mathbf{c}:N_{\mathbf{c}}\ni v}{f_{cv}} = \sum_{u\in N_v} f_{vu} \\ (3)\ \underset{\mathbf{c}:N_{\mathbf{c}}\ni v}{f_{cv}} \delta_s + \sum_{v'\in N_v} f_{vu}\delta_t + \sum_{u\in N_v} f_{uv}\delta_r \le \mathbf{B}_v \\ (4)\ f_{vu} \ge 0, \forall v, u\in\mathcal{V} \end{cases}$$

The above problem formulation is similar to the original one, except for some additional constraints on the virtual node. Constraint 1 ensures that the amount of generated data at each virtual node equals the outflow from that virtual node. Constraint 2 specifies the flow conservation for ordinary sensor nodes. Constraint 3 specifies the energy constraint on each sensor node. In fact, this problem is equivalent to the original problem in the sense that (1) any solution of the above problem can be converted to that of the original problem without utility degradation, and (2) both problems share the same objective function. In particular, after solving the above problem, we immediately obtain a rate allocation for the original problem by setting the sensing rate of v to $s_v = f_{\mathbf{c}v}$, and the amount of flow on link uv to f_{uv}. It is easy to verify that this allocation is feasible in satisfying both flow conservation and energy constraint. In the rest of this paper, we will study our problem based on CCG.

4.2 Optimal Fair Rate Allocation

It turns out to be extremely difficult to tackle the original QoM-aware rate allocation problem directly, therefore we start with a new problem, called *fair rate allocation* problem. *Fair rate allocation* problem is also known as the lexicographic maxmin rate allocation problem in [20]. Later, we show that the *fair rate allocation problem* and QoM-aware rate allocation problem share the common optimal solution under some settings. Different from the QoM-aware rate allocation problem, whose objective is to maximize the total utilities, summed over all components, *fair rate allocation* problem seeks a rate allocation which can maximize the minimum sensing rate among all components. Surprisingly, we show that the optimal solutions to those two problems are identical when $\delta_s \ge \delta_r$. In the rest of this section, we first introduce the *fair rate allocation* problem, and then provide an efficient algorithm to tackle this problem. Remember that we use $S = \{s_{\mathbf{c}_1}, s_{\mathbf{c}_2}, \cdots, s_{\mathbf{c}_m}\}$ to represent a rate allocation, where $s_{\mathbf{c}_i}$ is the rate assigned to component $\mathbf{c}_i$.

DEFINITION 3 (FAIR RATE ALLOCATION). *Given two feasible sensing rate allocations S_a and S_b, we sort them in non-decreasing order, and obtain two non-decreasing rate vectors Q_a and Q_b. Let Q_a^i and Q_b^i represent the i-th rate in Q_a and Q_b, respectively. We define $S_a = S_b$ if, and only if, $Q_a = Q_b$; $S_a > S_b$ if, and only if, there exists an i such that $Q_a^i > Q_b^i$ and for all $j < i$, $Q_a^j = Q_b^j$. We say a rate allocation S is an optimal* fair rate allocation *if, and only if, there exists no other rate allocation $S' > S$.*

In order to solve the *fair rate allocation* problem, we adopt a similar approach proposed in [20]. The difference between our problem setting and [20] is that we only care about the sensing rate of each correlation component instead of each individual sensor node. This approach is composed of two parts: (1) Maximum Common Rate Computation: compute a maximum common rate $\overline{s}$ that satisfies all energy constraints and flow conservation; and (2) Maximum Individual Rate Computation: calculate the maximum rate for each component by assuming the sensing rate of all the other components is $\overline{s}$. We compute those two rates iteratively, until the final rate allocation is determined.

I. COMPUTE MAXIMUM COMMON RATE. To compute the maximum common rate, we formulate it as a linear programming problem. In this problem, most constraints are the same as those listed in *QoM Aware Rate Allocation based on CCG*, except that (1) we use the same sensing rate $\overline{s}$ for all correlation components in Constraint 1, and (2) the objective is to find the maximum $\overline{s}$. We can use any linear programming solver, such as simplex methods, to efficiently solve this problem and obtain the maximum common sensing rate $\overline{s}$.

II. COMPUTE MAXIMUM INDIVIDUAL RATE. After the maximum common rate $\overline{s}$ is computed, the second step is to compute the maximum individual sensing rate that can be achieved for each component by assuming all the other components take the same sensing rate $\overline{s}$. For each component $\mathbf{c}$, we formulate the maximum individual rate problem as a linear programming problem. Its formulation is similar to the one defined in the previous phase, except that the first condition is replaced by two constraints (1) $s_{\mathbf{c}} = \sum_{v \in N_{\mathbf{c}}} f_{\mathbf{c}v}$; and (2) $s_{\mathbf{c}'} = \sum_{v \in N_{\mathbf{c}'}} f_{\mathbf{c}'v} = \overline{s}, \forall \mathbf{c}' \in \mathcal{C} \setminus \{\mathbf{c}\}$. Essentially, the objective is to compute the maximum $s_{\mathbf{c}}$ under the constraint that the remaining components have a common sensing rate $\overline{s}$. This problem can still be solved efficiently through linear programming. After obtaining the maximum individual rate for each component, there must exist at least one component, say $\mathbf{c}$, whose maximum individual rate is the same as the maximum common rate. We find all such $\mathbf{c}$, and set their final sensing rate to $s_{\mathbf{c}} = \overline{s}$.

The pseudo codes are listed in Algorithm 1. By solving the previous two problems iteratively, we can determine the rate allocation and associated flow assignment for each correlation component. Theorem 1 shows that the rate allocation returned from Algorithm 1 is an optimal *fair rate allocation*. The general idea of the proof follows that of [20], and the difference is that in our problem, as mentioned earlier, the sensing rate is associated with each correlation component instead of each individual sensor node.

Algorithm 1 Optimal Fair Rate Allocation (FRA)

Input: CCG and associated energy constraint & flow conservation.
Output: Sensing rate for each component and flow assignment on each link.

while $\mathcal{C} \neq \emptyset$ **do**
  Compute the maximum common sensing rate $\overline{s}$ in $\mathcal{C}$;
  for each component $\mathbf{c}$ in $\mathcal{C}$ **do**
    Compute the maximum individual sensing rate $s_{\mathbf{c}}$ by assuming the sensing rate of all other components is $\overline{s}$;
    if $s_{\mathbf{c}} = \overline{s}$ then $\mathcal{C} \longleftarrow \mathcal{C} - \mathbf{c}$
return the rate allocation.

THEOREM 1. *Algorithm 1 returns the optimal* fair rate allocation.

4.3 QoM-aware Rate Allocation for Utility Maximization

So far, we have demonstrated that Algorithm 1 returns the optimal solution in terms of *fair rate allocation* problem; however, there is still a gap between the optimal *fair rate allocation* and optimal QoM-aware rate allocation. To fill this gap, we show in Theorem 2 that if the per unit data sensing cost is no less than the per unit data receiving cost, then the optimal *fair rate allocation* is also an optimal QoM-aware rate allocation. It implies that, when $\delta_s \geq \delta_r$, Algorithm 1 returns the optimal QoM-aware rate allocation, which achieves the maximum total utility.

We first provide Lemma 1 as a supporting lemma of Theorem 2. Lemma 1 can be easily proved based on the definition of optimal *fair rate allocation*, as in Definition 3. In particular, Lemma 1 reveals an important property of optimal *fair rate allocation*; this property will be used later to establish the equivalence between optimal *fair rate allocation* and optimal QoM-aware rate allocation under some settings.

LEMMA 1. *Given any optimal* fair rate allocation*, in order to increase some correlation component's sensing rate (if possible), we must reduce the sensing rate of some other component who has a lower sensing rate.*

In the following, we prove that Algorithm 1 returns the optimal QoM-aware rate allocation under the assumption that the per unit data sensing cost is no less than the per unit data receiving cost.

THEOREM 2. *When the per unit data sensing cost is no less than the per unit data receiving cost $\delta_s \geq \delta_r$, Algorithm 1 returns the optimal QoM-aware rate allocation.*

PROOF. To better illustrate the big idea used to prove this theorem, we put our focus on a simple case, by assuming that the communication graph is a linear network where the sink is the leftmost node. Later, similar techniques can be extended to the proof for general communication graph.

PROPOSITION 1. *Given any feasible rate allocation S in the linear communication graph with $\delta_s \geq \delta_r$, in order to increase the sensing rate of some component, say $\mathbf{c}^*$, by ϵ, we only need to decrease the total sensing rate of the other components by at most ϵ.*

PROOF. See Figure 4 as an illustration. Given a feasible rate allocation S, we aim to increase the sensing rate of some component, say $\mathbf{c}^*$, by ϵ. For any rate adjustment strategy which can achieve this goal, let $\{v_i : i = 1, 2, \cdots, k\}$ be the set of all nodes (which are not adjacent to $\mathbf{c}^*$) whose sensing rates are decreased. Let ϵ_i represent the decreased rate of node v_i, and $\{v_1, v_2, \cdots, v_k\}$ are ordered in increasing order of their hop distance to the sink, *e.g.*, v_1 is the node closest to the sink.

We study two cases, respectively, in terms of different distributions of ϵ_i, and show that the total amount of decreased rate $\sum_{i=1}^{k} \epsilon_i$ is no larger than ϵ:

$\triangleright$ If $\sum_{i=2}^{k} \epsilon_i \geq \epsilon$, we can simply set $\epsilon_1 = 0$, without violating the energy constraint of v_1. Notice that, compared to the original rate allocation S, the increased flow from $\mathbf{c}^*$ to v_1 is ϵ. However, since we have $\sum_{i=2}^{k} \epsilon_i \geq \epsilon$, this indicates that the reduced flow from all the other nodes is no less than the increased flow from v_1. In other words, there is no additional flow coming into v_1, compared to S. Because S is a feasible allocation, the energy constraint of v_1 still holds. We perform this operation sequentially to v_1, v_2, $\cdots$ until it meets some node, say v_t, such that $\sum_{i=t}^{k} \epsilon_i > \epsilon$ and $\sum_{i=t+1}^{k} \epsilon_i < \epsilon$, then we apply the following operation.

$\triangleright$ If $\sum_{i=t}^{k} \epsilon_i > \epsilon$ and $\sum_{i=t+1}^{k} \epsilon_i < \epsilon$, we can set the decreased rate of v_t to $\sum_{i=t}^{k} \epsilon_i - \epsilon$ without violating the energy constraint of v_t. According to similar arguments used in the previous operation, we know that the overall increased flow through v_t is $\epsilon - \sum_{i=t+1}^{k} \epsilon_i$. Because the sensing cost is no less than the receiving cost $\delta_s \geq \delta_r$, therefore decreasing the sensing rate of v_t by at most $\epsilon - \sum_{i=t+1}^{k} \epsilon_i$, the energy constraint is still satisfied. After this stage, we can ensure that the total decreased rate is at most:

$$\epsilon - \sum_{i=t+1}^{k} \epsilon_i + \sum_{i=t+1}^{k} \epsilon_i = \epsilon$$

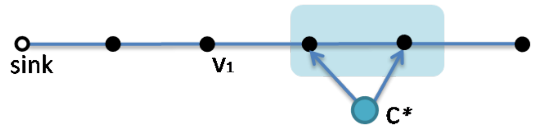

Figure 4: Linear communication graph where sink is the leftmost node. The objective is to increase the sensing rate of component $\mathbf{c}^*$ by ϵ. Given any rate adjustment strategy which can achieve this goal, assume v_1 is the leftmost node (which does not belong to $\mathbf{c}^*$) whose sensing rate should be decreased.

This completes the proof. □

LEMMA 2. *For the linear communication graph, any optimal* QoM-aware rate allocation *must also be an optimal* fair rate allocation *if* $\delta_s \geq \delta_r$.

PROOF. We prove this through contradiction. Recall that in Lemma 1, we demonstrate that given any optimal *fair rate allocation*, we cannot increase the sensing rate of a correlation component without reducing the sensing rate of some other correlation component with a lower sensing rate. Given an optimal QoM-aware rate allocation S, assume by contradiction that there exists a component $\mathbf{c}^*$ whose sensing rate $\delta_{\mathbf{c}^*}$ can be increased without decreasing the rate of any other component with a smaller rate. Notice that in Proposition 1, we show that, in order to increase the sensing rate of some component by some amount ϵ, we only need to decrease the sensing rate of the other components by, at most, the same amount ϵ. Therefore, we are able to obtain a new rate allocation S^* by (1) increasing $\mathbf{c}^*$'s sensing rate from $\delta_{\mathbf{c}^*}$ to $\delta_{\mathbf{c}^*}+\epsilon$, and at the same time (2) decreasing the total sensing rate of some other components with a higher rate, by at most ϵ. Recall that the utility function defined for each component is a submodular function with diminishing return property, thus if $s_{\mathbf{c}^*} < s_{\mathbf{c}_1} < s_{\mathbf{c}_2} < \cdots < s_{\mathbf{c}_k}$ and $\epsilon = \sum_1^k \epsilon_k$, we have

$$\mathcal{U}(s_{\mathbf{c}^*}+\epsilon)+\mathcal{U}(s_{\mathbf{c}_1}-\epsilon_1)+\mathcal{U}(s_{\mathbf{c}_2}-\epsilon_2)+\cdots+\mathcal{U}(s_{\mathbf{c}_k}-\epsilon_k)$$

$$\geq \mathcal{U}(s_{\mathbf{c}^*})+\mathcal{U}(s_{\mathbf{c}_1})+\mathcal{U}(s_{\mathbf{c}_2})+\cdots\mathcal{U}(s_{\mathbf{c}_k})$$

It indicates that the total utility of S^* is larger than that of S. This contradicts the assumption that S is an optimal QoM-aware rate allocation. □

A similar approach can be applied to the proof for general graph. In particular, we prove:

PROPOSITION 2. *Given any feasible rate allocation S in general communication graph with $\delta_s \geq \delta_r$, in order to increase the sensing rate of some component, say $\mathbf{c}^*$, by ϵ, we only need to decrease the total sensing rate of the other components by at most ϵ.*

PROOF SKETCH. The technique used to prove this proposition is similar to that used in Proposition 1, except for the selection of v_1. We sort all nodes in topological ordering, then instead of picking the leftmost node as v_1, we pick the node with the smallest order as v_1. The remainder of the proof follows a flow similar to that in Proposition 1. Essentially, we scan all nodes in increasing order of their topological ordering to find all those nodes whose sensing rate can be adjusted. The detailed proof is omitted here to save space.

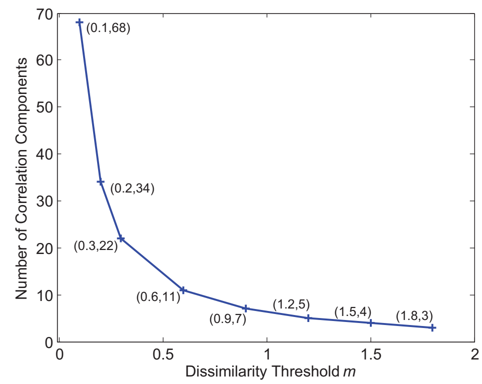

Figure 5: Magnitude dissimilarity threshold versus the number of correlation components.

Based on Proposition 2, and similar proof used in Lemma 2, we are able to show that for general communication graph, any optimal QoM-aware rate allocation is also an optimal *fair rate allocation* if $\delta_s \geq \delta_r$, and vice versa due the uniqueness of the optimal solution.

5. DISCUSSION AND FUTURE WORK

In this work, we study the QoM-aware rate allocation problem. In this section, we briefly discuss the limitations of current work, and further propose some interesting extensions as our future work.

► So far we assume that that the per unit data sensing cost is no less than the per unit data receiving cost. This requirement may not always be satisfied in practice, especially for those low energy cost sensors *e.g.*, accelerometer or gyro sensors. One interesting direction for future work is to develop a new rate allocation scheme whose performance can be bounded for the general case.

► In this paper, we assume that the utility function $\mathcal{U}()$ is identical to all correlation components; the other possible direction for future research is to generalize existing studies to heterogeneous utility functions, *e.g.*, different components may have different submodular utility functions. In this case, the current approach may fail to achieve the maximum QoM, even under the assumption that the per unit data sensing cost is no less than the per unit data receiving cost.

► Another interesting extension is to study the rate allocation problem in multi-application networks. In reality, one node may be involved in many applications [1] [26], such as temperature, humidity, and illuminance monitoring. For different applications, we may partition the network into different correlation components. As a result, one node may belong to multiple components under a multi-application network. Then one interesting problem is to study how to find a rate allocation among sensors and applications, which can achieve the maximum utility across all applications. One possible approach is to add additional virtual nodes for all new added components, and then apply similar approaches as proposed in this work.

6. PERFORMANCE EVALUATION

6.1 Experimental Results

Our outdoor testbed adopts TelosB Mote [27] with a MSP430 processor and CC2420 transceiver. Each mote is equipped with 2 AA batteries. The sensor program is developed based on TinyOS

2.1. The energy consumptions for receiving, transmitting, and sensing unit data are 21.8 mA, 19.5 mA and 22 mA respectively.

6.1.1 Utility Function

We decided to use a multivariate Gaussian, which is widely used in the body of literature [28], as an utility function. We then write our utility function for each component $\mathbf{c}$ in terms of total sensing rate $s_{\mathbf{c}}$ as $\mathcal{U}(s_{\mathbf{c}}) = \log s_{\mathbf{c}} A e^{-\left(\frac{1}{\sigma^2}\right)}$, where A is some constant, and σ is the variance of readings among sensor nodes within one component. It is easy to verify that $\mathcal{U}(s_{\mathbf{c}})$ is indeed a submodular function. The overall utility is defined as $\mathbf{U} = \sum_{\mathbf{c} \in \mathcal{C}} \mathcal{U}(s_{\mathbf{c}}) = \sum_{\mathbf{c} \in \mathcal{C}} \log s_{\mathbf{c}} A e^{-\left(\frac{1}{\sigma^2}\right)}$.

6.1.2 Dissimilarity Score Threshold and Its Impact

In order to exploit the correlation among the data reported by the sensor nodes, and to help reduce the redundant samplings, we dynamically partition the network into a set of disjoint correlation components. The sensor nodes within the same correlation component have strong correlation and, therefore, great similarity in sensing values. At any time instant, only a small fraction of sensor nodes need to be active, serving as the representatives for the whole correlation component. The partition operation is based on the dissimilarity measure, as used in [11]. The dissimilarity measure of time series is computed in a pairwise manner, based on historical observations from individual sensor nodes.

We define two metrics to evaluate the difference of the two time series, *magnitude dissimilarity* and geographic distance. Two time series $\{x_1, x_2, \cdots, x_t\}$ and $\{y_1, y_2, \cdots, y_t\}$ are *magnitude dissimilar* if there is an i $(1 \leq i \leq t)$ such that $|x_i - y_i| > \mathbf{m}$. Here $\mathbf{m}$ denotes the dissimilarity threshold. In our experiments, we put two sensors S_x and S_y into different correlation components if their time series are magnitude dissimilar, or their geographic distance is greater than a threshold value, which is set to be 30 feet.

The goal of this set of experiments is to explore the impact of the dissimilarity threshold value on the number of correlation components. By varying the magnitude dissimilarity threshold value $\mathbf{m}$, we collect a set of performance data, as illustrated in Figure 5. It demonstrates that, with the decrease of dissimilarity threshold value, the number of correlation components increases. This is not surprising, for the reason that a lower dissimilarity threshold value leads to a higher data resolution requirement. If $\mathbf{m} = 0$, each individual sensor node constitutes an independent component itself. On the one hand if $\mathbf{m} = \infty$, we treat the entire network as one component. Intuitively speaking, neither $\mathbf{m} = 0$ nor $\mathbf{m} = \infty$ is a good choice; the first setting completely ignores the potential correlation among sensor nodes, and the second setting somehow overestimates their correlation. As we will discuss later, different selections of $\mathbf{m}$ significantly affects the system performance in terms of estimation accuracy.

6.1.3 Performance in Estimation Error Minimization

An appropriate dissimilarity threshold value is essential to reducing the estimation error under a specific energy budget. The goal of this set of experiments is to find the appropriate dissimilarity threshold value that minimizes the estimation error, given an energy budget. We use the difference distortion measure, which has been broadly used in image compression, to evaluate the accuracy of a reconstructed image against the original image [29]. The difference distortion measure σ^2 is defined by $\sigma^2 = \frac{\sum_{j=1}^{M}\sum_{i=1}^{N}(S_{ij}-R_{ij})^2}{M \times N}$ where S_{ij} is the jth actual sensing value from the ith sensor node, and R_{ij} is the jth restoration value of the ith sensor node at the sink. N denotes the total number of sensor nodes, and M denotes the total number of samplings from each sensor node. We use a normalized difference distortion measure as the estimation error by normalizing σ^2 by the average variation of samplings. $\sigma^2_{norm} = \frac{N\sigma^2}{\sum_{i=1}^{N}\mathrm{Var}(S_i)}$ where $\mathrm{Var}(S_i)$ denotes the sensing value variation of the ith sensor node. In our experiment, we set the energy budget for each sensor node as 1 mA per unit time (*i.e.*, minute). We use a random sensing rate allocation (RSRA) protocol as a baseline for performance comparison. Under RSRA protocol, all sensing values are collected through CTP (Collection Tree Protocol) [30]. The sensing rate of each sensor is decided in a distributed manner, where each sensor randomly selects a sensing rate under its current energy budget. The remaining energy is used to relay data for its neighbors. We collect estimation errors under different dissimilarity threshold values and plot the results in Figure 6.

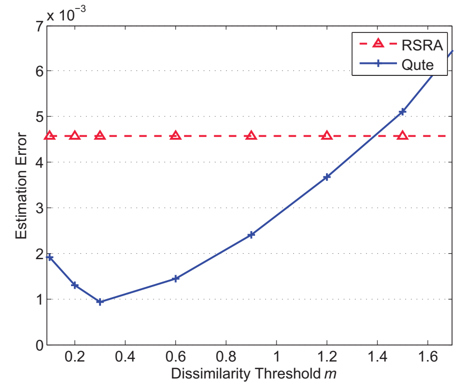

Figure 6: Magnitude dissimilarity threshold versus the estimation error.

As shown in Figure 6, the estimation error of the RSRA protocol does not change under different dissimilarity threshold values; this is because the rate allocation under RSRA protocol is irrelevant to the dissimilarity threshold. For Qute, the change of dissimilarity threshold values heavily affects the estimation accuracy of the temperature distribution recovered at the sink node. Specifically, a lower dissimilarity threshold value leads to a smaller correlation component. In an extreme case, each individual node stands for an independent component without considering correlation among sensor nodes, which clearly leads to poor estimation accuracy. On the other hand, a higher dissimilarity threshold value leads to misplacement of non-correlated sensor nodes into one correlation component, resulting in failure of collecting a high-resolution temperature distribution at the sink node. We observe from the experimental results that the estimation error is minimized when the dissimilarity threshold value is set to 0.3. Therefore, this value will be used as a default in the following experiments.

6.1.4 Performance in Utility Maximization

Using the utility function defined in Section 6.1.1, we perform a set of experiments to evaluate the performance of *Qute*, in terms of achieved utility under different energy budgets. By varying the energy budget from 0.1 mA per time unit to 8 mA per unit time, we collect a set of performance data, and plot the results in Figure 7. RSRA protocol is used as a baseline for comparison. As shown in the figure, *Qute* outperforms RSRA protocol in all energy budget values. The reason is that *Qute* computes the sensing rate assignment for each sensor node based on their correlation, while

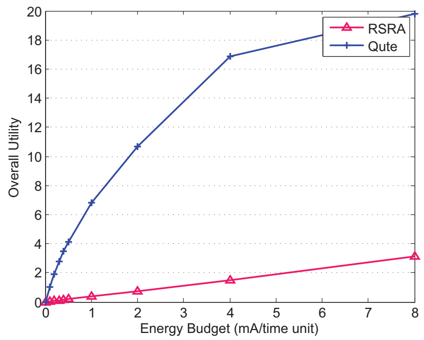

Figure 7: Energy budget versus the overall utility.

RSRA lets the sensor node randomly pick a sensing rate. With the same energy budget, *Qute* therefore achieves significantly higher overall utility by fully exploiting the correlation distribution in the network.

7. CONCLUSION

In this work, we study the QoM-aware rate allocation problem by exploring the correlational relationship among sensor nodes. We adopted an efficient rate allocation strategy, called *Qute*, by jointly optimizing both sensing rate selection and routing. We analytically show that *Qute* can achieve the maximum total utility if the sensing cost is no less than the receiving cost. In the future, we are interested in a more general problem setting, in which different components may have different utility functions.

8. REFERENCES

[1] Y. Xu, A. Saifullah, Y. Chen, C. Lu, and S. Bhattacharya, "Near optimal multi-application allocation in shared sensor networks," in *ACM MobiHoc*. ACM, 2010, pp. 181–190.

[2] S. Tang, X. Li, X. Shen, J. Zhang, G. Dai, and S. Das, "Cool: On coverage with solar-powered sensors," in *ICDCS*. IEEE, 2011.

[3] V. Delouille, R. Neelamani, and R. Baraniuk, "Robust distributed estimation in sensor networks using the embedded polygons algorithm," in *IPSN*. ACM, 2004.

[4] H. Gupta, V. Navda, S. Das, and V. Chowdhary, "Efficient gathering of correlated data in sensor networks," in *ACM MobiHoc*, 2005.

[5] S. Yoon and C. Shahabi, "Exploiting spatial correlation towards an energy efficient clustered aggregation technique (cag)[wireless sensor network applications]," in *ICC*, vol. 5. IEEE, 2005, pp. 3307–3313.

[6] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[7] M. Vuran, O. Akan, and I. Akyildiz, "Spatio-temporal correlation: theory and applications for wireless sensor networks," *Computer Networks*, 2004.

[8] S. Tang and L. Yang, "Morello: A quality-of-monitoring oriented sensing scheduling protocol in sensor networks," in *INFOCOM, 2012*. IEEE, 2012, pp. 2676–2680.

[9] S. Lauritzen, *Graphical models*. Oxford University Press, USA, 1996, vol. 17.

[10] L. Villas, A. Boukerche, D. Guidoni, H. Oliveira, R. Araujo, and A. Loureiro, "Time-space correlation for real-time, accurate, and energy-aware data reporting in wireless sensor networks," in *ACM MSWIM*, 2011.

[11] C. Liu, K. Wu, and J. Pei, "An energy-efficient data collection framework for wireless sensor networks by exploiting spatiotemporal correlation," *TPDS*, vol. 18, no. 7, pp. 1010–1023, 2007.

[12] V. Isler and R. Bajcsy, "The sensor selection problem for bounded uncertainty sensing models," *Automation Science and Engineering, IEEE Transactions on*, 2006.

[13] A. Krause, J. Leskovec, C. Guestrin, J. VanBriesen, and C. Faloutsos, "Efficient sensor placement optimization for securing large water distribution networks," *Journal of Water Resources Planning and Management*, 2008.

[14] F. Bian, D. Kempe, and R. Govindan, "Utility based sensor selection," in *IPSN*. ACM, 2006, pp. 11–18.

[15] S. Pradhan and K. Ramchandran, "Distributed source coding: Symmetric rates and applications to sensor networks," in *DCC 2000*.

[16] P. Von Rickenbach and R. Wattenhofer, "Gathering correlated data in sensor networks," in *Proceedings of the 2004 joint workshop on Foundations of mobile computing*. ACM, 2004, pp. 60–66.

[17] A. Scaglione and S. Servetto, "On the interdependence of routing and data compression in multi-hop sensor networks," *WINET*, vol. 11, no. 1-2, pp. 149–160, 2005.

[18] C. Wu, Y. Xu, Y. Chen, and C. Lu, "Submodular game for distributed application allocation in shared sensor networks," in *IEEE INFOCOM 2012*.

[19] I. Akyildiz, M. Vuran, and O. Akan, "On exploiting spatial and temporal correlation in wireless sensor networks," in *Intl. Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks 2004*.

[20] S. Chen, Y. Fang, and Y. Xia, "Lexicographic maxmin fairness for data collection in wireless sensor networks," *TMC*, vol. 6, no. 7, pp. 762–776, 2007.

[21] K.-W. Fan, Z. Zheng, and P. Sinha, "Steady and fair rate allocation for rechargeable sensors in perpetual sensor networks," in *ACM SenSys*. ACM, 2008.

[22] S.-J. Tang and J. Yuan, "On distributed sensing scheduling to achieve high quality of monitoring," in *2013 Proceedings IEEE INFOCOM*, Turin, Italy, Apr. 2013.

[23] L. Mo, Y. He, Y. Liu, J. Zhao, S.-J. Tang, X.-Y. Li, and G. Dai, "Canopy closure estimates with greenorbs: sustainable sensing in the forest," in *ACM SenSys*, 2009.

[24] A. Meka and A. Singh, "Distributed spatial clustering in sensor networks," *Advances in Database Technology 2006*.

[25] C. Detweiler, M. Doniec, M. Jiang, M. Schwager, R. Chen, and D. Rus, "Adaptive decentralized control of underwater sensor networks for modeling underwater phenomena," in *ACM SenSys*. ACM, 2010.

[26] A. Tavakoli, A. Kansal, and S. Nath, "On-line sensing task optimization for shared sensors," in *IPSN*, 2010.

[27] J. Polastre, R. Szewczyk, and D. Culler, "Telos: Enabling ultra-low power wireless research," in *IPSN*, 2005.

[28] D. Lynch, D. McGillicuddy *et al.*, "Objective analysis for coastal regimes," *Continental Shelf Research*, 2001.

[29] K. Sayood, *Introduction to data compression*. Morgan Kaufmann, 2005.

[30] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection tree protocol," in *ACM SenSys*. ACM, 2009.