On the Delay Performance of In-network Aggregation in Lossy Wireless Sensor Networks

Changhee Joo, *Member, IEEE,* and Ness B. Shroff, *Fellow, IEEE*

***Abstract*—In this paper, we study the implication of wireless broadcast for data aggregation in lossy wireless sensor networks. Each sensor node generates information by sensing its physical environment and transmits the data to a special node called the sink, via multi-hop communications. The goal of the network system is to compute a function at the sink from the information gathered by spatially distributed sensor nodes. In the course of collecting information, in-network computation at intermediate forwarding nodes can substantially increase network efficiency by reducing the number of transmissions. On the other hand, it also increases the amount of the information contained in a single packet and makes the system vulnerable to packet loss. Instead of retransmitting lost packets, which incurs additional delay, we develop a wireless system architecture that exploits the diversity of the wireless medium for reliable operations. To elaborate, we show that for a class of aggregation functions, wireless broadcasting is an effective strategy to improve delay performance while satisfying reliability constraint. We provide scaling law results on the performance improvement of our solution over unicast architecture with retransmissions. Interestingly, the improvement depends on the transmission range as well as the reliability constraint.**

***Index Terms*—Data aggregation, lossy wireless networks, delay performance.**

I. Introduction

Wireless sensor networks consist of a large number of sensor nodes with limited resources of energy, transmission power, network bandwidth, and computation power. Each sensor node monitors the physical environment in its neighborhood, collects data, and processes information. In many applications, the goal of wireless sensor networks is to compute a global function of the information gathered by spatially distributed sensors at a special node called the *sink*. Multi-hop communication is often used to relay the information from the source node to the sinks.

Distributed *in-network computation* (or *aggregation*) [1] can improve the communication efficiency of the system. It allows for an intermediate node to participate in the computation of the global function: a sensor node can collect information from a subset of sensors and aggregate it by performing computations with partial information. Compared with previous end-to-end information delivery paradigms, in which intermediate nodes simply relay the received information without change, distributed in-network computation can result in significant performance improvements in energy consumption, memory usage, bandwidth, and delay.

In this paper, we focus on the delay performance of in-network aggregation in lossy wireless networks. Under a noisy wireless channel, maintaining the overall reliability of the function computation while performing distributed computations at intermediate nodes is a major challenge [1]–[4]. Since the information contained in a single packet is highly intensified after several in-network computations, a packet loss can significantly impact the computation result, and thus a higher level of protection is required for each packet transmission. A packet can be protected by Error Correcting Code (*EEC*) [5] or can be restored by retransmitting the lost packet. In either case, additional delay is unavoidable. In many applications, it is important to compute the global function in a *timely* and *reliable* manner, and thus limiting the amount of additional delay is important.

To this end, we develop a new network architecture for in-network computation for a class of generalized maximum functions. We focus on the delay performance of the function computation subject to reliability constraint in lossy wireless environments. We show that aggregation with wireless broadcast can substantially reduce the delay while satisfying the reliability constraint. Our scaling law result clarifies the relationship between delay performance, reliability, and transmission range. We also provide distributed algorithms and evaluate their performance through simulations.

In-network aggregation has also been studied in many other aspects [1]. The maximum achievable computation rate for a class of functions has been investigated in [6]–[8]. Energy efficiency in lossy environments has been considered in [3], [4], [9]. Time and energy complexity of distributed computation has been provided in [10], [11]. Our work can be differentiated from the previous work in that i) we focus on the delay performance of in-network computation, ii) we consider reliability constraints in lossy wireless networks, and iii) we investigate the effect of wireless broadcast on the delay performance.

The paper is organized as follows. We first describe the system model in Section II. We provide scaling law results of asymptotic delay bounds under different reliability constraints and transmission ranges in Section III. In Section IV, we develop distributed algorithms to implement in-network aggregation that exploits wireless broadcasting in the presence of interference. In Section V, we evaluate our schemes through simulations. Finally, we conclude our paper in Section VI.

Manuscript received XXXX; revised XXX. This work was supported in part by NSF Awards CNS-1012700 and CNS-0905408, and by the Army Research Office MURI awards W911NF-07-10376 (SA08-03), and W911NF-08-1-0238, and in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education, Science, and Technology (No. 2011-0008549).

Earlier version of this work has been presented at the 46-th Annual Allerton Conference on Communications, Control, and Computing, Monticello, IL, USA, 2008.

Changhee Joo is with the School of Electrical and Computer Engineering, UNIST, Ulsan 689-798, Korea (email: cjoo@unist.ac.kr)

Ness B. Shroff is with the Depts. of ECE and CSE, The Ohio State University, Columbus, OH 43210 USA (email: shroff@ece.osu.edu)

II. System Model

We consider a sensor network $G(V,E)$ having a set V of sensor nodes and a set E of links, in which n sensor nodes are deployed. The goal is to compute a global function with information obtained from each sensor node. We assume that the function should be calculated at a special node, called the sink. Each sensor node not only generates its own data but also relays others data to the sink via multi-hop wireless communications. The wireless channel is assumed to be lossy. A packet loss can be restored by retransmitting the lost packet, which, however, results in additional delay. Since many applications have both reliability and delay constraints, we focus on the relationship between the reliability and the delay performance and show how they can improve when in-network aggregation is appropriately employed in the wireless system.

We are interested in a class of functions that satisfy all the following three properties.

- *Symmetric*: A function f is symmetric if $f(\vec{x},\vec{y}) = f(\vec{y},\vec{x})$.
- *Decomposable*: A function f is decomposable if $f(\vec{x},\vec{y}) = f(f(\vec{x}), f(\vec{y}))$.
- *Componentwise Transitive*: A function f is componentwise transitive if $[f(\vec{x},\vec{y})]_i = [f(\vec{x})]_i$ and $[f(\vec{y},\vec{z})]_i = [f(\vec{y})]_i$ imply that $[f(\vec{x},\vec{z})]_i = [f(\vec{x})]_i$, where $[\cdot]_i$ denote the i-th element of the vector.

We denote this class of functions by *Generalized Maximum* (GM) functions, since the final result corresponds to an element (could be a vector element) of the sensed data. Some examples include $\max$ or $\min$, ranging (i.e., $[\min,\max]$), and n-largest (or smallest) values. Many sensor network services can be realized through this class of functions: intrusion detection by collecting binary information, object tracking by collecting n-closest locations to the object and their distances, and multi-modal environmental monitoring (e.g., finding highest temperature with humidity exceeding a certain threshold) [12], [13]. Also, in general wireless networks, this type of functions might need to be calculated frequently to update system parameters such as the largest node degree, the longest queue length, the worst link quality, etc [14], [15].

The properties of the GM functions promote in-network aggregation. Specifically, an intermediate node can collect information from other sensor nodes, and instead of directly relaying the received packets, it processes and aggregates them into a unit of information, i.e., a packet. It then forwards the computed value to the sink or to the next hop. Appropriate use of in-network aggregation can significantly reduce the amount of traffic generated over the network [3], [4], [6]. Another important feature of the GM functions is that they allow duplicate data, i.e., inserting another copy of data does not affect the function results. We actively exploit this feature to battle against lossy wireless channels.

Our model is based on the following assumptions.

Assumption 1. The information generated at each sensor node is exact without error.

Assumption 2. The message passing computation model [11] is assumed, i.e., all the information has to be explicitly transmitted and silence periods (including listening to others' activities) cannot be used to convey information. Hence, if a sensor node does not transmit a packet, its information cannot contribute to the global function computation.

Assumption 3. Time is slotted (each slot is equal to a sampling period) and all sensor nodes are assumed to be synchronized. Scheduling is perfect in TDMA network systems.

Assumption 4. Routing is fixed. We first consider a tree topology rooted at the sink, which is a popular structure in wireless sensor networks because information heads for a fusion center (sink). We also modify the topology later to incorporate wireless broadcast.

Assumption 5. The wireless channel between each pair of transmitting and receiving nodes is assumed to be independent across links and times, and modeled as a binary channel with non-zero packet loss probability p. To avoid trivialities, We assume that p is bounded as $0 < \underline{p} \leq p \leq \overline{p} < 1$.

At time slot t, each sensor node ν generates information β_ν by sensing its physical environments. Our objective is to calculate a GM function value $f(\beta_1, \beta_2, \ldots, \beta_n)$ at the sink, that conveys the aggregated information from the sensor nodes in a timely and reliable manner. Let β^* denote the correct function value that has to be reported, i.e., $\beta^* := f(\beta_1, \ldots, \beta_n)$. The information value of β_j is said to be *critical* if the function result without β_j is different from β^*, i.e, $f(\beta_1, \ldots, \beta_{j-1}, \beta_{j+1}, \ldots, \beta_n) \neq \beta^*$. For instance, let $f(\cdot) = \min\{\cdot\}$ and $\beta_1 = 5$, $\beta_2 = 2$, and $\beta_3 = 9$. Then β_2 has the critical information value because $f(\beta_1, \beta_3) = 5 \neq 2 = f(\beta_1, \beta_2, \beta_3)$. Note that if the information is represented by a vector with m components, there can be at most m critical information values, since the component-wise transitive property implies that a single critical information determines a component of the function result. In the sequel, for easy of exposition, we assume that $m = 1$, and $f(\cdot)$ is a $\max$ function with a single element. However, since the three properties allow the critical information to be duplicated and to be aggregated in any order and in any intermediate node, our analysis can be easily extended to any GM function with $m > 1$, and as long as m is a constant, our asymptotical results do not change. Let $\hat{\beta}$ denote the information of the critical value. Since $\beta^* = \hat{\beta}$ for $m = 1$, we use β^* and $\hat{\beta}$ interchangeably in the remainder of the paper.

III. Asymptotic Analysis of the Delay Bound

Let P_s denote the minimum probability that the sink computes the function correctly. We study the asymptotic delay performance of the sensor system for the following reliability constraint:

$$1 - P_s = O\left(\frac{1}{c(n)}\right), \tag{1}$$

i.e., there exists $n_0, c_0 > 0$ such that for all $n > n_0$, $1 - P_s \leq \frac{c_0}{c(n)}$, where $c(n)$ is an increasing function of n with $c(n) \to \infty$ as $n \to \infty$, indicating the speed of convergence rate at which reliability is achieved as a function of the number of nodes.

A. Aggregation with unicast

We first consider a point-to-point communication system with a tree topology [11], where a node has a parent and multiple children (except the root node and leaf nodes). Each node obtains information in two ways: from its own sensor and from its children. Once a node collects information from all its children, it aggregates the information including its own into a single packet using the GM function and transmits the packet to its parent over a point-to-point communication link (unicast). The procedure repeats from leaf nodes to the root. We call this network architecture as aggregation with unicast and denote it by $\mathbb{U}$.

Since routes follow the tree structure rooted at the sink, each node ν has a unique parent $\mu(\nu)$. Let p denote the probability of loss for transmission over link (ν, $\mu(\nu)$). Let $r_u(n) \geq 0$ denote the maximum number of retransmissions allowed at each link, and let $P_s(\nu)$ denote the probability of success over link (ν, $\mu(\nu)$) when success occurs by taking the maximum number of allowed retransmissions. We can obtain $P_s(\nu)$ as

$$\begin{aligned} P_s(\nu) &= 1 - \text{Pr}\ \{\text{all transmissions fail}\} \\ &= 1 - p^{1+r_u(n)}. \end{aligned} \tag{2}$$

We define the depth $d(\nu)$ as the number of hops between node ν and the sink. Let $d^*(n)$ denote the maximum depth over all sensor nodes, i.e., $d^*(n) := \max_{\nu \in V} d(\nu)$. Then, P_s, the worst-case probability of success, is given by the success probability that the information of the critical value arrives at the sink through the longest path. Letting $\hat{\nu}$ denote the node that generates the information $\hat{\beta}$ of the critical value, we have

$$P_s = \min_{\hat{\nu}=\nu;\nu \in V} \prod_{k=1}^{d(\hat{\nu})} P_s(\nu_k) = \prod_{k=1}^{d^*(n)} P_s(\nu_k), \tag{3}$$

where $\nu_1 := \hat{\nu}$ and $\nu_{k+1} := \mu(\nu_k)$ for all $k > 1$. The last equality holds because in the worst-case, $\hat{\nu}$ has the largest depth $d^*(n)$. By substituting (2) into (3), we can obtain the following inequality:

$$c_1 \cdot d^*(n) \cdot \underline{p}^{1+r_u(n)} \leq 1 - P_s \leq c_2 \cdot d^*(n) \cdot \overline{p}^{1+r_u(n)}, \tag{4}$$

where c_1 and c_2 are some constant. From the left side of (4) and from the reliability constraint of (1), we have

$$1 + r_u(n) \geq c_3 \cdot \log(d^*(n) \cdot c(n)) + c_4,$$

for some constant c_3 and c_4. Also, from the right side of (4), the inequality with some constant is sufficient for the reliability constraint. Hence, a scheme that satisfies the reliability constraint (1) should have

$$r_u(n) \geq \Theta(\log(d^*(n) \cdot c(n))), \tag{5}$$

and the bound is tight in the sense that some scheme with the equality can satisfy the reliability requirement.

We now consider the delay caused by retransmissions to achieve the given reliability constraint. Estimating the delay by the number of transmissions, the worst-case delay D_u^* can be presented as $D_u^* = \min_{r_u(n)}\{d^*(n)\cdot(1+r_u(n))\}$. From (5), we have $\min(1+r_u(n)) = \Theta(\log(d^*(n)\cdot c(n)))$, which implies that a packet should be transmitted at least $\Theta(\log(d^*(n)\cdot c(n)))$ times to satisfy the reliability constraint. Hence, we obtain the *worst-case* delay under aggregation with unicast as

$$D_u^* = \Theta(d^*(n) \cdot \log(d^*(n) \cdot c(n))). \tag{6}$$

B. Aggregation with wireless broadcast

In this section, we propose a new network architecture with wireless broadcast to improve the delay performance while achieving the same level of reliability. We explicitly exploit diversity from wireless broadcast. We first describe the system architecture and then analyze its delay performance.

We modify the tree structure in Section III-A by allowing nodes to broadcast a packet to *multiple parents*.

Assumption 4.1. Each node (at depth d) has at least $x(n)$ parents[1] (at depth $d-1$), and transmits a packet through the wireless broadcast channel to all parents (1+$r_b(n)$) times. At the root, we assume that the sink has $x(n)$ antennas and it can process signals from multiple antennas.

In this architecture, we say that a node *successfully transmits* a packet if the broadcasted packet is successfully received by one of $x(n)$ parents. Note that each packet contains aggregated information abstracting all the information successfully collected by the transmitter. Due to the properties of the GM functions, it is sufficient that each node successfully transmits the aggregated information to *one of its parents* in order to ensure that the information $\hat{\beta}$ of the critical value is successfully delivered to the sink. We call this architecture as aggregation with broadcast and denote it by $\mathbb{B}$.

The intuition can be better described using Fig. 1. Assuming that links are bidirectional, the dotted lines in the figure is a link between two nodes, and arrows indicate a transmission from a child to a parent. A failed transmission is marked by a cross. Fig. 1(a) illustrates that two transmissions from node 2 fail under $\mathbb{U}$. On the other hand, Fig. 1(b) shows that a single broadcast can transmit the packet to node 6 successfully. Hence, $\mathbb{U}$ requires four transmissions to deliver information B to the sink, whereas $\mathbb{B}$ needs two transmissions.

Note that the aggregation with broadcast $\mathbb{B}$ appears to be a little like *flooding*, but there are significant differences. While flooding is very ineffective because of broadcasting multiple duplicate packets, $\mathbb{B}$ reduces this inefficiency by in-network aggregation. Moreover, it takes advantage of the diversity of wireless broadcast, which is not exploited in flooding.

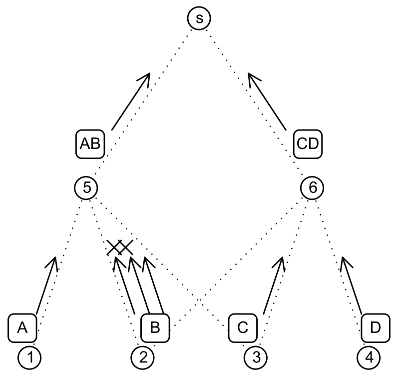
We now estimate the worst-case probability P_s of successful function computation under $\mathbb{B}$. Assuming independent packet losses over links (Assumption 5), a packet transmission from node ν is successful with probability

$$P_s(\nu) \geq 1 - \overline{p}^{x(n)\cdot(1+r_b(n))}, \tag{7}$$

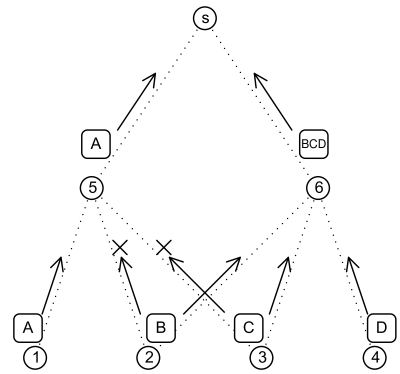
where $r_b(n)$ is the maximum number of retransmissions. Again, since the information $\hat{\beta}$ of the critical value has to be delivered via at most $d^*(n)$ hops to reach the sink, the guaranteed probability P_s of a successful information delivery can be represented by

$$P_s \geq \min_{\hat{\nu}=\nu;\nu \in V} \prod_{k=1}^{d(\hat{\nu})} P_s(\nu_k) = \prod_{k=1}^{d^*(n)} P_s(\nu_k), \tag{8}$$

[1]This implies that there are at least $x(n)$ disjoint paths from a node to the sink. Since there are $x(n)$ different first-hop nodes from the node's parents and each of these first-hop nodes has $x(n)$ parents, we can find at least $x(n)$ disjoint two-hop paths. Then by induction, we can show that there are at least $x(n)$ disjoint paths to the sink.



(a) Aggregation with unicast ($\mathbb{U}$)



(b) Aggregation with wireless broadcast ($\mathbb{B}$)

Fig. 1. Transmissions over lossy wireless links. Each transmission is denoted by an arrow, and a failed transmission is denoted by a cross at the end of the arrow. Under aggregation with unicast, it needs four transmissions for information B to be successfully delivered to the sink, while it needs two transmissions under aggregation with broadcast.

where $\nu_1 := \hat{\nu}$ and ν_{k+1} is one of parents of ν_k. Note that unlike $\mathbb{U}$, the first equality in (3) is changed to an inequality in (8) because the information $\hat{\beta}$ can take multiple (at least $x(n)$) paths to the sink. From (7), we can obtain

$$1 - P_s \le d^*(n) \cdot \overline{p}^{x(n)(1+r_b(n))}. \tag{9}$$

Then for $c_5 := \frac{-1}{\log \overline{p}}$ and a constant c_6, the following inequality suffices to satisfy the reliability constraint (1):

$$x(n) \cdot (1 + r_b(n)) \ge c_5 \cdot \log(d^*(n) \cdot c(n)) + c_6. \tag{10}$$

Note that if each node broadcasts its packet to $c_5 \log(d^*(n) \cdot c(n)))$ parents, the reliability constraint (1) can be satisfied with $r_b(n) = 0$. Since the delay bound D_b^* can be represented as $D_b^* = \min_{r_b(n)}\{d^*(n) \cdot (1 + r_b(n))\}$, we have

$$D_b^* \le \Theta\left(d^*(n) \cdot \max\{1, \tfrac{\log(d^*(n) \cdot c(n))}{x(n)}\}\right). \tag{11}$$

C. Performance in geometric networks

We now consider a popular scenario in which sensor nodes are randomly deployed in a geometric space, and evaluate the delay performance of aggregation schemes with unicast $\mathbb{U}$ and with broadcast $\mathbb{B}$. We derive the gain of $\mathbb{B}$ over $\mathbb{U}$ for geometric networks, where the reliability constraint and transmission range are a function of the number of nodes. We show that in general a higher gain can be achieved with a stronger reliability requirement and a larger transmission range.

We first start with the gain for the previous (non-geometric) tree network. We define the maximum delay gain of $\mathbb{B}$ over $\mathbb{U}$ as $G^* := D_u^*/D_b^*$. From (6) and (11), we have[2]

$$\begin{aligned} G^* := \frac{D_u^*}{D_b^*} &= \Omega\left(\frac{d^*(n) \cdot \log(d^*(n) \cdot c(n))}{d^*(n) \cdot \max\{1, \frac{\log(d^*(n) \cdot c(n))}{x(n)}\}}\right) \\ &= \Omega\left(\frac{x(n) \log(d^*(n) \cdot c(n))}{x(n) + \log(d^*(n) \cdot c(n))}\right). \end{aligned} \tag{12}$$

Suppose that the network has depth $d^*(n) = \log n$ with the reliability constraint $c(n) = \log n$. From (6) and (11), we have $D_u^* = \Theta(\log n \cdot \log\log n)$ under $\mathbb{U}$, and $D_b^* \le \Theta(\log n \cdot \max\{1, \log\log n/x(n)\}) = \Theta(\log n)$ under $\mathbb{B}$ when $x(n) = \Theta(\log\log n)$ and $r_b(n) = O(1)$. Hence, if each node can broadcast to $\Theta(\log\log n)$ parents, $\mathbb{B}$ outperforms $\mathbb{U}$ by $G^* = \Omega(\log\log n)$.

However, the achievability of $x(n) = \Theta(\log\log n)$ depends on the topology of the underlying network. In geometric networks, both $d^*(n)$ and $x(n)$ are related to the topological structure and we need to incorporate some topological notion into our analysis. To this end, we study the delay performance of aggregation schemes in random networks, where nodes are uniformly placed, subject to reliability constraint. In our analysis, we do not take into account edge effects, assuming that all nodes have the same order of parent nodes[3]. Note that in sensor networks, most traffic heads for the sink. Hence, by carefully locating the sink, there would be few transmissions on the edge of the network. The assumption can be further supported by our development of a distributed algorithm in Section IV.

Assumption 4.2. Given a network of n sensor nodes uniformly and independently distributed on a disk of radius 1, each node has an identical transmission range $t(n)$ and has the same order of parents $x(n)$ The sink is located at the center with $x(n)$ antennas. Straight-line routing has been employed, thus achieving $d^*(n) = \frac{1}{t(n)}$, and all the paths from a node to the sink have asymptotically the same length. In the next section, we show that this can be achieved by a simple routing scheme.

Under aggregation with unicast $\mathbb{U}$, the delay bound directly comes from (6). By replacing $d^*(n)$ with $\frac{1}{t(n)}$, we have

$$D_u^* = \Theta\left(\frac{1}{t(n)} \log \frac{c(n)}{t(n)}\right). \tag{13}$$

On the other hand, under aggregation with broadcast $\mathbb{B}$, we have $x(n) \le \Theta(nt(n)^2)$ because each node has $nt(n)^2$ neighboring nodes in its transmission range. We can achieve the equality by setting the parents of each node to the set of

[2] $f(n) = \Omega(g(n))$ means that there exists constants $\bar{n}, \bar{c} > 0$ such that for all $n \ge \bar{n}$, $f(n) \ge \bar{c} g(n)$.

[3] If the nodes are uniformly distributed in space, they asymptotically have the same order of neighbors [16]. Then, as shown in Section IV, it is not hard to develop a scheme, under which each node asymptotically have the same order of parents.

nodes within a sector of its transmission range (to the direction of the sink). Then, from (11) and $d^*(n) = \frac{1}{t(n)}$, we can obtain the delay bound as

$$D_b^* \leq \Theta\left(\max\left\{\frac{1}{t(n)}, \frac{1}{nt(n)^3}\log\frac{c(n)}{t(n)}\right\}\right). \tag{14}$$

From (13) and (14), we can present the gain G_{geo}^* of $\mathbb{B}$ over $\mathbb{U}$ in geometric networks as

$$G_{\text{geo}}^* := \frac{D_u^*}{D_b^*} = \Omega\left(\frac{nt(n)^2 \cdot \log\frac{c(n)}{t(n)}}{nt(n)^2 + \log\frac{c(n)}{t(n)}}\right). \tag{15}$$

As an example, we consider a random geometric network with minimal connectivity. It has been shown in [17] that $t(n)$ should be at least $\Theta(\sqrt{\frac{\log n}{n}})$ for the network to be asymptotically connected with high probability. Using $t(n) = \Theta(\sqrt{\frac{\log n}{n}})$, we have $d^*(n) = \Theta\left(\sqrt{\frac{n}{\log n}}\right)$. Suppose that $c(n) = \log n$, i.e., the reliability requirement enforces that $1 - P_s = O(\frac{1}{\log n})$. In this case, the delay bound of $\mathbb{U}$ can be written as $D_u^* = \Theta(\sqrt{n \log n})$ from (13). For $\mathbb{B}$, it suffices to satisfy $x(n)(1 + r_b(n)) = \Theta\left(\log\left(\frac{n \cdot c(n)}{\log n}\right)\right)$ to achieve the same level of reliability. Since each node can have $x(n) = \Theta(nt(n)^2) = \Theta(\log n)$ parents, the condition can be satisfied with $r_b(n) = O(1)$ when $c(n) = \log n$. Further, if $x(n) \geq c_5 \log n$, there is no need of retransmission under $\mathbb{B}$. Hence, we can obtain the delay bound $D_b^* = O\left(\sqrt{\frac{n}{\log n}}\right)$ and the gain $G_{\text{geo}}^* = \Omega(\log n)$. Note that $nt(n)^2 = \log n$ is the number of nodes in the transmission area of a node. This implies that $\mathbb{B}$ can potentially achieve a gain in delay as large as the diversity gain of wireless broadcast.

In general, from (15), the gain depends on both $t(n)$ and $c(n)$. We tabulate the gains for various network environments in Table I. The first column shows that the gain is dominated by the broadcasting areas in multi-hop networks with minimal connectivity. The last column shows that the gain is dominated by the reliability constraint in single-hop networks. The results also show that we can improve the delay performance by exploiting multicast transmissions, with a smaller transmission range when a low level of reliability is required (i.e., when $c(n) \leq n$), and with a larger transmission range when a high level of reliability is required (i.e., when $c(n) \geq n$).

TABLE I
GAINS (D_u^*/D_b^*) OF $\mathbb{B}$ OVER $\mathbb{U}$ UNDER VARIOUS TRANSMISSION RANGES AND RELIABILITY CONSTRAINTS.

	$t(n) = \sqrt{\frac{\log n}{n}}$	$t(n) = \frac{1}{\sqrt{\log n}}$	$t(n) = 1$
$c(n) = \log n$	$\Omega(\log n)$	$\Omega(\log\log n)$	$\Omega(\log\log n)$
$c(n) = n$	$\Omega(\log n)$	$\Omega(\log n)$	$\Omega(\log n)$
$c(n) = \exp n$	$\Omega(\log n)$	$\Omega(\frac{n}{\log n})$	$\Omega(n)$

IV. DISTRIBUTED ALGORITHMS

In this section, we develop a practical solution for aggregation with broadcast using a tiered routing structure. Although the tiered structure has appeared in the literature for light-weight routing [18] and efficient sleep/wake scheduling [19], [20], the purpose of our design is quite different. Unlike [18]–[20], we assume that wireless links are lossy, and that the network has a specific goal of computing a GM function. By exploiting the diversity of the wireless medium, we intend to improve the delay performance while satisfying reliability constraint.

We first describe our solution, and show that the algorithm achieves the delay performance of (14). To this end, we show that under the algorithm, each sensor node has at least $\Theta(nt(n)^2)$ parents and the maximum hop distance to the sink is at most $\Theta(\frac{1}{t(n)})$. We extend our schemes to resource constrained networks, and revisit performance analysis in the presence of wireless interference. We close this section with development of a hybrid scheme that can combine the unicast and the broadcast architecture.

A. Algorithm with tiered structure

We assume that n wireless sensor nodes are uniformly deployed over a disk of radius 1. Our results can be extended to more general networks of different sizes and topologies, which impact on our analysis by a constant factor and do not affect our scaling results. Under Assumption 4.2, each node has an identical transmission range of $t(n)$, and we divide the networks into $\frac{1}{\delta t(n)}$ circular tiers as shown in Fig. 2, with $0 < \delta < 1$. Each tier has an identical width of $\delta t(n)$. Let T_i denote the set of nodes in the i-th tier, which is an area within distance of $(\delta t(n) \cdot (i-1), \delta t(n) \cdot i]$ from the sink. The sink is the only node in T_0.

The network is a time-slotted TDMA system. At the beginning of each time slot, each sensor node generates a packet with the sensed information. A time slot is further divided into mini-slots and in each mini-slot, a single packet can be transmitted. Let D_b denote the delay performance of the algorithm, which is estimated in the number of transmissions, i.e., mini-slots for the sink to compute the function.

Routing is simplified using the tiered structure; Every node μ in T_i is a parent of node ν in T_{i+1} if its distance is no greater than $t(n)$. Transmissions are scheduled from the outermost tier to the sink tier-by-tier one at a time, so that nodes in T_i can transmit only after all nodes in T_{i+1} finish their transmissions. We group nodes in each tier into mutually exclusive subsets such that all nodes in a subset can transmit simultaneously. Let $H(i,j)$ denote the j-th subset in T_i, and let h_i denote the total number of subsets in each tier T_i such that $\cup_{j=1}^{h_i} H(i,j) = T_i$. Clearly, all nodes in T_i can finish a *single* transmission in h_i mini-slots. If there is no interference between simultaneous transmissions within a tier, we will have a single group with $h_i = 1$ for all tier i. The reason that we introduce grouping will become clearer in the next section when we take into account wireless interference. We will have the total delay D_b to compute the function as

$$D_b = \sum_{i=1}^{1/\delta t(n)} h_i(1 + r_b(n)). \tag{16}$$

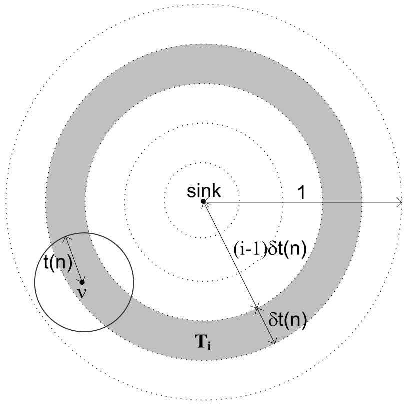


Fig. 2. Network with tiered structure.

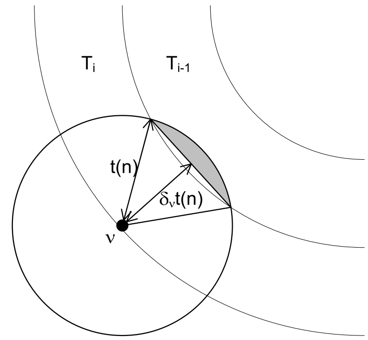


Fig. 3. Parents (in T_{i-1}) of node ν (in T_i) is located in the shaded area.

Algorithm 1 Distributed aggregation with wireless broadcast.

for $i = \frac{1}{\delta t(n)}$ to 1 **do**
  for $j = 1$ to h_i **do**
    Each node ν in $H(i,j)$ broadcasts its (aggregated) information $(1 + r_b(n))$ times.
    if node $\mu \in T_{i-1}$ receives the packet **then**
      Node μ does aggregation and updates its information.
    end if
  end for
end for

The overall algorithm proceeds as follows: At the beginning of every time slot, each node originates a packet with sensed information. Nodes in $H(i,j)$ broadcast their packets $(1 + r_b(n))$ times in decreasing order of i and increasing order of j, such as $H(\frac{1}{\delta t(n)}, 1), H(\frac{1}{\delta t(n)}, 2), \ldots, H(\frac{1}{\delta t(n)}, h_{\frac{1}{\delta t(n)}}), H(\frac{1}{\delta t(n)} - 1, 1), H(\frac{1}{\delta t(n)} - 1, 2), \ldots, H(1, h_1)$. Note that with this ordering, nodes in T_i start their transmissions after all nodes in T_{i+1} finish transmissions. Then nodes in T_i who receive a packet from a node in T_{i+1} do aggregation using the GM function, and update their packet if necessary. The detailed algorithm is as shown in Algorithm 1.

Now we show that under Algorithm 1, each node has at least $\Theta(nt(n)^2)$ parents and the maximum hop distance from a node to the sink is $\Theta(\frac{1}{t(n)})$. Then the minimum number of parents $x(n)$ can be bounded as follows. Suppose that node ν is located in T_i as shown in Fig. 3. The number of parents of node ν in T_{i-1} is no smaller than the number of nodes in the shaded area. For each node $\nu \in V$, there exists $\delta < \delta_\nu < 1$ such that the distance between ν and the shaded area is $\delta_\nu t(n)$. Let $\delta^* := \max_{\nu \in V} \delta_\nu$. Since nodes are uniformly distributed with density $\frac{n}{\pi}$, it can be easily shown that the number of nodes in the shaded area is bounded below by[4]

$$x(n) \geq \left(\frac{\cos^{-1} \delta^*}{\pi} - \delta^* \sqrt{1 - {\delta^*}^2} \right) \frac{nt(n)^2}{\pi} = \Theta(nt(n)^2). \tag{17}$$

[4]Although we implicitly assume that the shaded area is completely included in T_{i-1}, the same order results can be obtained when the shaded area stretches to inner tiers.

Since each node has at least $\Theta(nt(n))^2$ parents, Algorithm 1 can achieve the required reliability (1) by satisfying (10) with some $r_b(n) = \Theta(1)$.

Further, since each tier has the width $\delta t(n)$ and a packet is transmitted tier-by-tier, there are at most $\frac{1}{\delta t(n)}$ tiers and we have the maximum number of hops to the sink as

$$d^*(n) = \frac{1}{\delta t(n)} = \Theta\left(\frac{1}{t(n)}\right). \tag{18}$$

Hence, from (17) and (18), Algorithm 1 achieves the delay performance (14) and the gain (15) with some $r_b(n) = \Theta(1)$. However, this can be achieved only when there is no interference between simultaneous transmissions and all the nodes in each tier i belong to the same group with $h_i = 1$.

B. *Performance in the presence of interference*

In Section III, we have analyzed the performance (e.g., (15) and Table I) without considering wireless interference. However, if the network is resource-constrained and has limited frequency channels, then wireless interference will restrict the number of simultaneous transmissions, e.g., h_i of Algorithm 1, and this has to be factored into calculation of the gains.

Assumption 6. We consider a protocol model for the interference constraints [16], where two links within two times of transmission range cannot transmit simultaneously.

Multiple nodes within a tier can transmit simultaneously if the distance between any two of them is greater than $2t(n)$. We show that $h_i = \Theta(nt(n)^2)$, and obtain the delay performance of Algorithm 1 in the presence of wireless interference.

We first analyze the delay performance of Algorithm 1 by providing an algorithm that multiple nodes in a tier can be scheduled without interference. Then we compare the solution with a realization of $\mathbb{U}$, which appears in [11] with $t(n) = \sqrt{\frac{\log n}{n}}$ in a lossless network, and extended accordingly. We evaluate their performance and clarify the improvement of $\mathbb{B}$ over $\mathbb{U}$ in different network settings.

From (16), we need to estimate h_i to obtain D_b, which is determined by the scheduling policy within a tier. To this end, we first estimate $|H(i,j)|$, where $H(i,j)$ is the subset of nodes in T_i that are scheduled simultaneously, and $|\cdot|$ denotes the cardinality of the set. We partition T_i into subsets $\{C_i^m\}$ as shown in Fig. 4. Note that since each cell C_i^m has a width more than $t(n)$ at the boundary of the inner tier, there are at

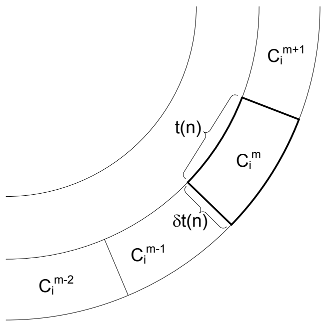


Fig. 4. Partition $\{C_i^m\}$ of T_i.

most $\lceil \frac{2\pi(i-1)\delta t(n)}{t(n)} \rceil$ cells, where $\lceil a \rceil$ is the closest integer no smaller than a. Let $H(i,j)$ include a node from every three cells, i.e., $H(i,j)$ has a node from cells $C_i^m, C_i^{m+3}, \ldots$, so on. Since any two nodes in $H(i,j)$ are separated more than $2t(n)$, they do not have any common parent and their transmissions do not interfere with each other. Moreover, since all cells has the same number of nodes (possibly except one cell, which may have a smaller number of nodes), the number of nodes in each $H(i,j)$ is identical and would be about a third of the number of C_i^m. Specifically,

$$|H(i,j)| = \left\lfloor \frac{1}{3} \left\lceil \frac{2\pi(i-1)\delta t(n)}{t(n)} \right\rceil \right\rfloor = \lfloor \tfrac{1}{3} \lceil 2\pi(i-1)\delta \rceil \rfloor, \quad (19)$$

for all $i > 1$, where $\lfloor a \rfloor$ is the closest integer no greater than a. The number of nodes in T_i can be bounded by

$$2\pi(i-1)\delta t(n) \cdot \delta t(n) \cdot \frac{n}{\pi} \leq |T_i| \leq 2\pi i \delta t(n) \cdot \delta t(n) \cdot \frac{n}{\pi}, \quad (20)$$

for all $i > 1$. Since nodes are uniformly distributed, we can obtain from (19) and (20) the number of mini-slots h_i needed for all nodes in T_i to make a single transmission as

$$h_i = \Theta\left(\frac{|T_i|}{|H(i,j)|}\right) = \Theta(nt(n)^2),$$

for all $i > 1$. For $i = 1$, it is clear $h_1 = \Theta(nt(n)^2)$ because $|T_1| = n\delta^2 t(n)^2$. Hence, we have $h_i = \Theta(nt(n)^2)$ for all i. From (10) and (16), we obtain

$$\begin{aligned} D_b &= \sum_{i=1}^{1/\delta t(n)} h_i(1 + r_b(n)) \\ &= \Theta\left(\frac{1}{\delta t(n)} \cdot nt(n)^2 \cdot (1 + r_b(n))\right) \qquad (21) \\ &= \Theta\left(nt(n)\left(1 + \frac{1}{nt(n)^2} \log \frac{c(n)}{t(n)}\right)\right). \end{aligned}$$

Remarks: Intuitively, each tier has width $\Theta(t(n))$ and thus includes $\Theta(nt(n))$ nodes. Since we can schedule a set of nodes, where distance between any two nodes is no smaller than $t(n)$, the number of scheduled nodes will be at most $\Theta(1/t(n))$. Hence, it takes at least $\Theta(nt(n)^2)$ mini-slots to finish transmissions *in each tier*. We can obtain the above equation by multiplying the term $nt(n)^2$, which explains the wireless interference, to (14).

Now we consider the realization of $\mathbb{U}$ presented in [11]. The algorithm is designed in lossless networks with minimal transmission range for connectivity, i.e., $t(n) = \sqrt{\frac{\log n}{n}}$, and shown to be optimal. We extend it into lossy networks with general transmission range $t(n)$ as follows:

1) Among sensor nodes, there are $\Theta(\frac{1}{nt(n)^2})$ nodes who locally collect information from its neighbors and do aggregation. They can be placed such that they form a tree with depth $\Theta(\frac{1}{t(n)})$ and nodes of the same depth do not interfere with each other.
2) At the beginning of each time slot, each node transmits its packet over a point-to-point communication link to the nearest collecting node. Due to retransmissions for lost packets, it takes $\Theta(nt(n)^2(1 + r_u(n)))$ times.
3) After the above procedure, all information is now located in collecting nodes. Then, each collecting node, starting from leaf node, transmits packet to its immediate parent up to $(1 + r_u(n))$ times. After receiving all packets from children, each collecting node does aggregation and transmits the data to its parents. This procedure takes $\Theta(\frac{1}{t(n)}(1 + r_u(n)))$ times until all information arrives at the sink.

From the above and (5), the algorithm has the delay performance

$$\begin{aligned} D_u &= \Theta\left(\left(nt(n)^2 + \frac{1}{t(n)}\right)(1 + r_u(n))\right) \\ &= \Theta\left(\left(nt(n)^2 + \frac{1}{t(n)}\right) \log \frac{c(n)}{t(n)}\right). \end{aligned} \qquad (22)$$

Note that the wireless interference is incorporated in the first term. Unlike Algorithm 1, it is added to (13) instead of being multiplied. This is because the interference matters only when nodes transmit packets to collecting nodes. On the other hand, in Algorithm 1, the interference remains through the procedure because it continuously exploits the wireless broadcast.

From (21) and (22), the gain can be obtained as

$$G = \Theta\left(\frac{(1 + nt(n)^3) \log \frac{c(n)}{t(n)}}{nt(n)^2 + \log \frac{c(n)}{t(n)}}\right).$$

Table II summarizes the gain of Algorithm 1 over the instance of $\mathbb{U}$ in the presence of wireless interference for various network settings. Algorithm 1 outperforms the instance of $\mathbb{U}$ in most cases. However, in some cases, e.g., $c(n) = \log n$ and $t(n) = \frac{1}{\sqrt{\log n}}$, the instance of $\mathbb{U}$ has better delay performance than Algorithm 1. Such a case occurs when either of the following two conditions holds:

1) $\log \frac{c(n)}{t(n)} < \frac{1}{t(n)}$, if $t \geq \frac{1}{\sqrt[3]{n}}$,
2) $\log \frac{c(n)}{t(n)} < nt(n)^2$, if $t \leq \frac{1}{\sqrt[3]{n}}$.

Note that poor delay performance could be caused either by a limited number of simultaneous transmissions due to wireless interference or by a large number of retransmissions required for reliability. Although Algorithm 1 exploits user and path diversity improving delay performance by reducing the number of retransmissions, the improvement may not be significant due to wireless interference. Table II shows that when the transmission range is small, Algorithm 1 does not perform very well since the delay from interference dominates.

In contrast, when the transmission range is large, Algorithm 1 can improve the delay performance substantially, while the gain depends on the reliability constraint. The results imply that broadcasting is more useful when a larger transmission range is required, e.g., due to topological restriction or delay deadline of sensed data.

TABLE II
GAINS OF ALGORITHM 1 OVER AN INSTANCE OF $\mathbb{U}$ UNDER VARIOUS WIRELESS STRUCTURES AND RELIABILITY CONSTRAINTS.

	$t(n) = \sqrt{\frac{\log n}{n}}$	$t(n) = \frac{1}{\sqrt{\log n}}$	$t(n) = 1$
$c(n) = \log n$	$\Theta(1)$	$\Theta(\frac{\log\log n}{\sqrt{\log n}})$	$\Theta(\log\log n)$
$c(n) = n$	$\Theta(1)$	$\Theta(\sqrt{\log n})$	$\Theta(\log n)$
$c(n) = \exp n$	$\Theta(1)$	$\Theta(\frac{n}{(\sqrt{\log n})^3})$	$\Theta(n)$

It is also worthwhile noting that when there is no loss in links and $t(n) = \sqrt{\frac{\log n}{n}}$, Algorithm 1 has the delay of $\Theta(\sqrt{n \log n})$ and the instance of $\mathbb{U}$ achieves $\Theta(\sqrt{\frac{n}{\log n}})$. Hence, the instance of $\mathbb{U}$ is better in lossless networks with minimal connectivity.

C. A hybrid scheme

Motivated by the cases that the instance of $\mathbb{U}$ outperforms the instance of $\mathbb{B}$, we consider a hybrid method that blends $\mathbb{U}$ and $\mathbb{B}$. Under the hybrid scheme, the information obtained by an individual sensor node is collected by some special nodes called collecting nodes, and these collecting nodes are responsible for data aggregation and information delivery to the sink. The scheme seems similar to $\mathbb{U}$, but there are important differences in that all packet transmissions are done by wireless broadcast and that (uniformly distributed) collecting nodes can interfere with each other. We first describe the implementable algorithm and provide a sufficient condition to satisfy the reliability constraint. Then, we analyze the delay performance and the gain of the hybrid method.

We use the tiered structure of Algorithm 1.

Assumption 4.2.1 In addition to Assumption 4.2, we further assume that among all sensor nodes, there are collecting nodes that are uniformly deployed over the network. Each node has at least $y(n) \in [1, nt(n)^2]$ collecting nodes within its transmission area.

The algorithm consists of two phase:

1) Phase 1: Each non-collecting node broadcasts its packet $(1 + r_{b1}(n))$ times. All nearby collecting nodes receive the packet and do aggregation.
2) Phase 2: From the outermost tier, each collecting node in T_i broadcasts its packet $(1+r_{b2}(n))$ time. Collecting nodes in T_{i-1} receive the packet and do aggregation. This procedure repeats tier-by-tier as Algorithm 1.

Remarks: The algorithm has some similarity with the solution in [11], which, however, operates with unicast, does not take into account packet losses, and requires specific placement of collecting nodes.

Sufficient condition for the reliability constraint:

Letting P_{s1} and P_{s2} denote the probability of successful packet transmission in phase 1 and at each tier in phase 2, respectively. The probability P_s of successful delivery of critical information value can be written as

$$\begin{aligned} P_s &\geq P_{s1} \cdot \prod_{k=1}^{d^*(n)} P_{s2} \\ &\geq (1 - \overline{p}^{y(n)\cdot(1+r_{b1}(n))}) \cdot (1 - \overline{p}^{y(n)\cdot(1+r_{b2}(n))})^{d^*(n)} \\ &\geq 1 - \overline{p}^{y(n)\cdot(1+r_{b1}(n))} - d^*(n) \cdot \overline{p}^{y(n)\cdot(1+r_{b2}(n))}. \end{aligned}$$

To simplify equations, we drop (n) in the sequel. Using $d^* = \Theta(\frac{1}{t})$, we obtain

$$\begin{aligned} &1 - P_s \leq \Theta\left(\tfrac{1}{c}\right), \\ &\text{if } y \cdot (1 + r_{b1}) \geq \Theta(\log c) \text{ and } y \cdot (1 + r_{b2}) \geq \Theta(\log \frac{c}{t}). \end{aligned} \tag{23}$$

Hence, $y\cdot(1+r_{b1}) \geq c_5 \log c$ and $y\cdot(1+r_{b2}) \geq c_5 \log \frac{c}{t}$ with $c_5 = \frac{-1}{\log \overline{p}}$ (and some $r_{b1} = \Theta(1), r_{b2} = \Theta(1)$) are sufficient conditions to satisfy the reliability constraint (1).

Delay performance:

Let D_h denote the delay bound of the hybrid scheme. Letting D_{h1} and D_{h2} denote the delay incurred by phase 1, and the delay incurred by phase 2, respectively, we have $D_h = D_{h1} + D_{h2}$.

Using the techniques provided in the previous sections so far and from (23), the following can be easily shown

$$\begin{aligned} D_{h1} &= \Theta\left(nt^2 \cdot \left(1 + \frac{1}{y}\log c\right)\right), \\ D_{h2} &= \Theta\left(\frac{y}{t} \cdot \left(1 + \frac{1}{y}\log \frac{c}{t}\right)\right), \end{aligned}$$

where in D_{h1}, the term nt^2 is the time for all non-collecting nodes to broadcast a packet and the following $(1 + \frac{1}{y}\log c)$ is required for retransmissions, and in D_{h2}, the term $\frac{y}{t}$ is the time for collecting nodes (y) in a single tier to broadcast a packet multiplied by the number of tiers ($\frac{1}{t}$), and the following $(1 + \frac{1}{y}\log \frac{c}{t})$ is required for retransmissions. Then we obtain

$$D_h = \Theta\left(nt^2 + \frac{y}{t} + \frac{nt^2}{y}\log c + \frac{1}{t}\log \frac{c}{t}\right). \tag{24}$$

The gain G^h of the hybrid scheme of the instance of $\mathbb{U}$ can be presented from (22) as

$$G^h := \frac{D_u}{D_h} = \Theta\left(\frac{\left(nt^2 + \frac{1}{t}\right)\log \frac{c}{t}}{nt^2 + \frac{y}{t} + \frac{nt^2}{y}\log c + \frac{1}{t}\log \frac{c}{t}}\right). \tag{25}$$

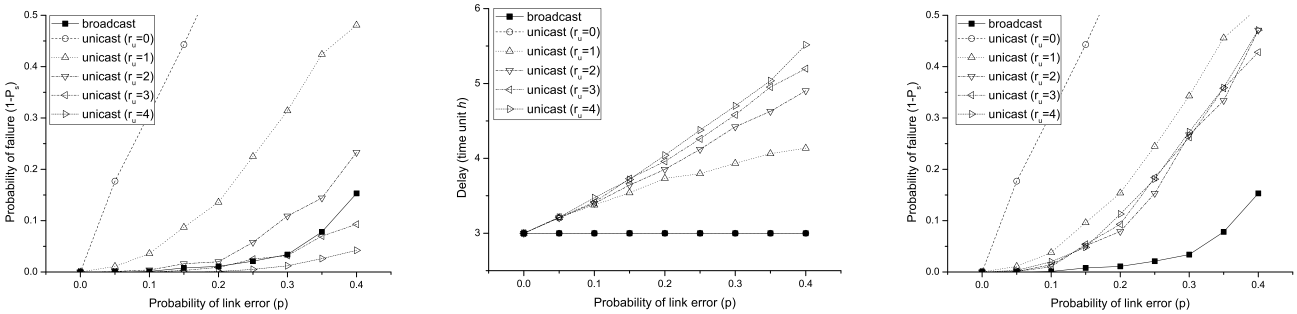
We denote $\mathcal{G}, \mathcal{N}, \mathcal{D}$ respectively, as

$$\mathcal{G} = \frac{\mathcal{N}}{\mathcal{D}} := \frac{\left(nt^2 + \frac{1}{t}\right)\log \frac{c}{t}}{nt^2 + \frac{y}{t} + \frac{nt^2}{y}\log c + \frac{1}{t}\log \frac{c}{t}}.$$

Differentiating both sides by y, we obtain

$$\frac{d\mathcal{G}}{dy} = \frac{\mathcal{N}}{\mathcal{D}^2} \cdot \left(\frac{nt^2}{y^2}\log c - \frac{1}{t}\right).$$

Note that the sign of $\frac{d\mathcal{G}}{dy}$ is determined by $\left(\frac{nt^2}{y^2}\log c - \frac{1}{t}\right)$,



(a) Loss rate

(b) Delay

(c) Loss rate with delay constraint.

Fig. 5. Loss rate and delay of information delivery. The loss rates of aggregation with unicast improves with the number of retransmissions r_u in Fig. 5(a). However, the increase of r_u leads to higher delay performance in Fig. 5(b). In contrast, there is no retransmission under the broadcast-based scheme ($r_b = 0$) and under the unicast-based scheme with $r_u = 0$. Thus their delays remain constant regardless of the loss rates. When accounting for information delivery within delay bound (min + 2 unit times, i.e., $2\hat{h}$), broadcasting without retransmissions shows better performance than unicast with retransmissions.

which is a monotonically decreasing function of $y \in [1, nt^2]$. Hence, $\frac{d\mathcal{G}}{dy} < 0$ if $\left(\frac{nt^2}{y^2}\log c - \frac{1}{t}\right)|_{y=1} < 0$, which implies that $\mathcal{G}$ also monotonically decreases in $[1, nt^2]$, and thus can be maximized when $y = 1$. Similarly, $\frac{d\mathcal{G}}{dy} > 0$ if $\left(\frac{nt^2}{y^2}\log c - \frac{1}{t}\right)\Big|_{y=nt^2} > 0$, and $\mathcal{G}$ can be maximized when $y = nt^2$. Otherwise, $\mathcal{G}$ will be maximized when $\left(\frac{nt^2}{y^2}\log c - \frac{1}{t}\right) = 0$, which leads to the setting of $y = \sqrt{nt^3 \cdot \log c}$. Summarizing, we obtain the optimal setting for the hybrid scheme as

$$y(n) = \begin{cases} \Theta(1), & \text{if } \log c < \frac{1}{nt^3}, \\ \Theta(nt^2), & \text{if } \frac{1}{t} < \log c, \\ \Theta(\sqrt{nt^3 \cdot \log c}), & \text{if } \frac{1}{nt^3} \leq \log c \leq \frac{1}{t}. \end{cases} \quad (26)$$

Therefore, the optimal density of collecting node depends on the reliability constraint and the transmission range (or the distance between a node and the sink). The exact gain is also determined by the choice of t, c, and y from (25). Since one of the four terms of (24) will dominate the others, the gain can present as

$$G^h(t, c, y) = \begin{cases} \Theta((1 + \frac{1}{nt^3}) \log \frac{c}{t}), \\ \Theta(\frac{1}{y}(1 + nt^3) \log \frac{c}{t}), \\ \Theta(y(1 + \frac{1}{nt^3}) \log \frac{c}{t} / \log c), \\ \Theta(1 + nt^3), \end{cases}$$

where the cases depend on t, c, y.

Since $c \to \infty$ and $t \in [\sqrt{\frac{\log n}{n}}, 1]$, we have $\Theta(\log \frac{c}{t}) > \Theta(1)$ and $\Theta(\log \frac{c}{t} / \log c) \geq \Theta(1)$. Then for all four cases, we achieve $G^h \geq \Theta(1)$ if $y \in [1, \Theta(nt^2)]$ is chosen accordingly. This result is expected: Since the instance of $\mathbb{U}$ is equivalent to the hybrid scheme with $y = 1$, the performance of the hybrid scheme with optimal parameter y must be no smaller than that of the instance of $\mathbb{U}$. Further, if $t(n) > \frac{1}{\sqrt[3]{n}}$, a gain strictly greater than $\Theta(1)$ will be achieved.

V. Simulation Results

In this section, we simulate our solutions, and evaluate their performance. We are interested in reliability in terms of successful transmissions as well as the delay. We first simulate scenarios of TDMA networks without interference, and proceed to resource-constrained networks with wireless interference.

A. TDMA networks without interference

We compare the performance of unicast-based and broadcast-based schemes in a wireless sensor network with 100 nodes, which are randomly placed in a disk of radius 1. The transmission range of each node is set to 0.5. The tiered structure has the width 0.25 ($\delta = \frac{1}{2}$), and a parent-child relationship has been established between every pair of nodes if the two nodes are located in neighboring tiers and their distance is less than 0.5. In this setting, there are four tiers. We locate the sensor node that generates the critical information values at the boundary of the network, i.e., in the 4-th tier. Since routing follows the tiered structure, both aggregations with unicast and broadcast takes at least four transmissions for the packet generated from the sensor node to arrive at the sink. We assume that for all tiers i, it takes the same number of mini-slots $\hat{h}$ for all nodes in T_i to finish a single transmission, and consider $\hat{h}$ as a time unit for the delay performance. For aggregation with unicast, we change the number of retransmissions r_u from 0 to 5, and for aggregation with broadcast we set $r_b = 0$. All links are assumed to fail transmission with the same probability p. Changing p, we count the number of time units ($\hat{h}$) required for the sink to receive the critical information value and measure the rate of failure, i.e., loss of the information. We run each simulation 1000 times and average the results.

Fig. 5 illustrates the loss rate of the critical information value and the delay performance. Fig. 5(a) shows that aggregation with unicast can improve the loss rate with more retransmissions. However, it also increases the delay as shown in Fig. 5(b). In contrast, there is no retransmission under the broadcast-based scheme ($r_b = 0$) and under the unicast-based scheme with $r_u = 0$. Thus their delays remain constant regardless of the loss rates. If we have the delay bound of $6\hat{h}$, which is the minimum achievable delay plus $2\hat{h}$, then it is observed in Fig. 5(c) that the retransmission strategy cannot

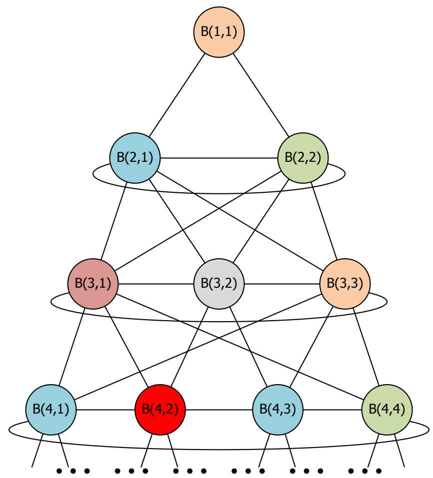


Fig. 6. Conflict graph of blocks. Each blocks contains 10 nodes. If a node in a block transmits, no other node in the connected blocks can transmit at the same time. Each vertex that represents a block is colored such that no two connected vertices have the same color.

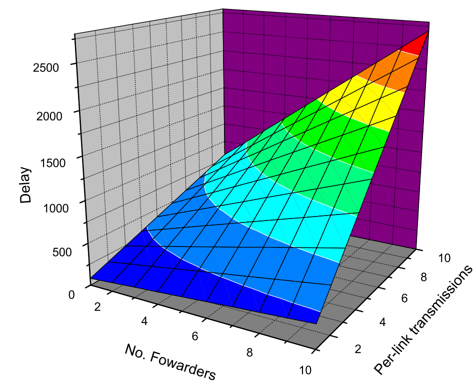


Fig. 7. Delay performance (time slots) in presence of interference, with different number of forwarders $k = 1, \ldots, 10$, and different number of per-link transmissions $r = 1, \ldots, 10$. Loss probability p does not affect the delay performance. Delay increases more quickly with increase in r than with increase in k.

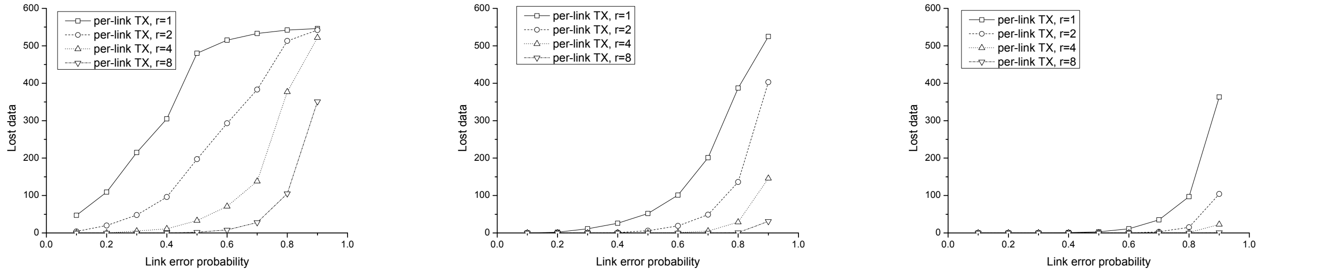
improve the loss rate beyond a certain threshold, and that aggregation with broadcast achieves better performance.

B. Resource-constrained networks with interference

We evaluate our hybrid schemes of Section IV taking into account wireless interference. The difficulty in the simulations lies in implementing an optimal scheduler. Since transmission time $\hat{h}$ in a tier changes with the number of collecting nodes, we need detailed implementation of scheduling functionality, which however often requires high computational complexity even under a very simple interference model. To facilitate implementation of the scheduling component, we consider the following block-based network, which captures essential features of wireless interference in tiered networks.

- **Network topology:** We group 10 nearby nodes as a block (like C_i^m). Nodes in a block are within communication range of each other, and they cannot transmit simultaneously due to interference constraints. Two blocks are *connected* when transmission of any node in a block can be received by all nodes in the other node. Also, we assume that no two nodes in the connected blocks can transmit simultaneously due to wireless interference.
  We assume that blocks have a tiered structure. There is only one block of nodes that can transmit to the sink, and this block consists of the first tier T_1. In the second tier T_2, there are two blocks, each of which is connected to the block in T_1, They are also connected with each other. Similarly, we assume that there are i blocks in each T_i. Let $B(i, j)$ denote the j-th block in T_i. For in-tier interference, we assume that $B(i, j)$ is connected with $B(i, j-1)$ and $B(i, j+1)$, where the addition and the substraction is modular-$(i+1)$ operation for the circular property of the tier, i.e., $B(i, 1)$ is connected with $B(i, i)$. For data forwarding and inter-tier interference, we connect each $B(i, j)$ to two blocks in T_{i-1}: to $B(i-1, j-1)$ and $B(i-1, j)$, where the substraction is again modular-i operation. We further assume that nodes in the block of the first tier are directly wired to the sink and hence, transmissions at the last hop, i.e., from nodes in T_1 to the sink, are neither lost nor interfere with other transmissions. We consider a network with total 10 tiers and 55 blocks.
- **Interference:** We can draw an equivalent conflict graph by representing a block as a vertex. A vertex $B(i, j)$ has an edge with vertices of $B(i, j-1)$ and $B(i, j+1)$ in tier i (sibling blocks), $B(i-1, j-1)$ and $B(i-1, j)$ in tier $i-1$ (parent blocks), and corresponding blocks in tier $i+1$ (child blocks). Assuming that there is no interference between non-connecting blocks, the interference relationship can be described in a simple form [5] as 'any intended transmitting node in a block should be the only transmitter within the block and its connected blocks.' Fig. 6 illustrates the conflict relationship among blocks.
- **Data transmission:** We assume a time-slotted system, where each time slot is further divided into mini-slots. Data is generated at the beginning of each time slot, and transmitted to the sink during the mini-slots in two steps: collecting and forwarding. In each block, we choose k out of 10 nodes as a collecting node (also denoted by a forwarder). First, each non-collecting node in a block broadcasts its data to all the collecting nodes in the block. Then the collecting nodes aggregate the received data and transmit to collecting nodes in the upper tier (i.e., to nodes in the parent blocks. Note that in our network structure, each collecting node in T_i have total $2k$ parent nodes in T_{i-1}.) It is an instance of broadcast model $\mathbb{B}$ if $k = 10$, and it is close to an instance of unicast model $\mathbb{U}$ if $k = 1$.
- **Scheduling:** For collecting data within a block, we schedule as follows. We first color blocks using 6 colors such

[5]The model takes into account wireless interference at the sender side, and does not consider interference at the receiver side.



(a) $k = 1$ (b) $k = 3$ (c) $k = 6$

Fig. 8. Lost information (number of sensor nodes) in the presence of interference, with different link loss probabilities $p = 0.1, \ldots, 0.9$, and different numbers of per-link transmissions $r = 1, 2, 4, 8$. Results with different numbers of forwarders $k = 1, 3, 6$ show that a small increase of forwarders can significantly improve reliability.

that no two connected blocks have the same color[6] as shown in Fig. 6. We use 3 colors at each tier. Nodes in the blocks of the same color, one node per block, can transmit at the same time without interference. Hence, it takes $6 \cdot (10 - k)$ mini-slots for each non-collecting node to transmit once. We assume that the nodes retransmit (i.e., re-broadcast) $r_x (\geq 0)$ times for reliable collecting. After collecting the data within blocks, the aggregated data is forwarded to the sink tier-by-tier. Note that blocks of the same color can transmit at the same time. Hence, all the collecting nodes in a tier can finish a transmission for $3k$ mini-slots. Nodes retransmit $r_y (\geq 0)$ times for forwarding.

We simulate our schemes changing the number of forwarders k from 1 to 10. Each link between two nodes has loss probability p, which changes in the range of $[0.1, 0.9]$. The number of (re)transmissions r per link also changes from 1 to 10, i.e., $1 + r_x = 1 + r_y = r \in [1, 10]$.

Fig. 7 illustrates the delay performance for the sink to get all the data under different k and r in terms of mini-slots. Link loss probability p does not affect the delay. The results show sharp increases in delay when the number of retransmissions r per link increases than when the number of forwarders k increases.

Fig. 8 shows the impact of forwarders on reliability. We measure the number of lost information with different link loss probabilities, numbers of retransmissions, and numbers of forwarders. The results show that a small number of forwarders significantly improve the reliability, especially when the link loss probability is high, i.e., under a harsh environment like under-water scenarios.

The gains of wireless broadcast are more visible in Fig. 9, which presents the delay and the data loss for the given number of forwarders. For each k forwarders, the lowest-delay point is a result when there is no retransmission (i.e., $r = 1$), the the next lowest-delay point is a result when there is 1 per-link retransmission, and so on. As the number of retransmissions increases, the reliability improves while the delay performance deteriorates. The curve for k forwarders can be considered as an achievable performance boundary with different number

[6]Indeed, it is sufficient with 5 colors in our particular case. If the number of blocks does not increase by one per tier, we may need 6 colors.

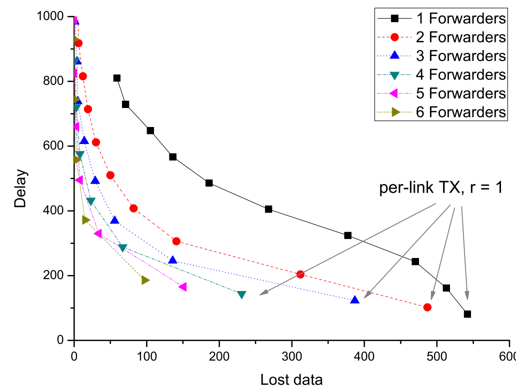


Fig. 9. Delay performance and the number of lost information in the presence of interference, with different numbers of forwarders $k = 1, \ldots, 6$, and different numbers of per-link transmissions $r = 1, \ldots, 10$. Loss probability $p = 0.8$. The performance boundary for each k forwarders improves approaching the origin, when the number of forwarders k increases.

retransmissions. The results show that the boundary improves, i.e., gets closer to the origin, as the number of forwarders increase.

VI. Conclusion

In a wireless sensor network, in-network aggregation can significantly improve efficiency when the goal of the network is to compute a global function. However, since the loss of an aggregated packet is far more harmful than an unaggregated packet, a higher level of protection is required for reliable operations in lossy wireless environments. In this paper, we use wireless broadcast as a means of protecting the aggregate information for a class of generalized maximum functions. Exploiting the diversity of wireless medium, broadcasting spreads information spatially, and the properties of the function enable distributed in-network computation with the spread information. We show that aggregation with broadcast can improve delay performance while satisfying the same level of reliability. The gain can be presented as a function of reliability constraint and transmission range.

Using a tiered network topological structure, we develop solutions for aggregation that exploit wireless diversity, and are amenable to implementation in a distributed manner. We evaluate the schemes in a resource-constrained network with wireless interference. Further, we also develop a hybrid scheme that combine the unicast and multicast architecture. Simulation results show that aggregation with broadcast outperforms aggregation with unicast, especially, in severely lossy network environments.

There are many interesting open questions to consider. Aggregation functions besides the generalized maximum functions should be considered. An open question is whether the performance bounds in the presence of interference in Section IV-B are tight or not. Although we focus on the delay performance, other performance metrics such as time complexity and achievable sampling rate are also of importance. It would be interesting to study the relationship between these metrics with aggregation functions and network topologies.

REFERENCES

[1] A. Giridhar and P. R. Kumar, "Toward a Theory of In-Network Computation in Wireless Sensor Networks," *IEEE Communications Magazine*, vol. 44, no. 4, pp. 98–107, April 2006.

[2] R. G. Gallager, "Finding Parity in a Simple Broadcast Network," *IEEE Trans. Inf. Theory*, vol. 34, no. 2, pp. 176–180, Mar 1988.

[3] E. Kushilevitz and Y. Mansour, "Computation in noisy radio networks," *SIAM J. Discret. Math.*, vol. 19, no. 1, pp. 96–108, 2005.

[4] L. Ying, R. Srikant, and G. Dullerud, "Distributed Symmteric Function Computation in Noisy Wireless Sensor Networks," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, December 2007.

[5] L. L. Peterson and B. S. Davie, *Computer Networks: A Systems Approach, 3rd Edition*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2003.

[6] A. Giridhar and P. R. Kumar, "Computing and Communicating Functions Over Sensor Networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 755–764, 2005.

[7] S. Kamath and D. Manjunath, "On Distributed Function Computation in Structure-Free Random Networks," in *ISIT*, July 2008.

[8] C. Li and H. Dai, "Towards Efficient Designs for In-network Computing With Noisy Wireless Channels," in *IEEE INFOCOM*, March 2010.

[9] J. Zhao, R. Govindan, and D. Estrin, "Computing Aggregates for Monitoring Wireless Sensor Networks," in *IEEE International Workshop on Sensor Network Protocols and Applications*, May 2003, pp. 139–148.

[10] J.-Y. Chen, G. Pandurangan, and D. Xu, "Robust Computation of Aggregates in Wireless Sensor Networks: Distributed Randomized Algorithms and Analysis," in *IPSN*, 2005, p. 46.

[11] N. Khude, A. Kumar, and A. Karnik, "Time and Energy Complexity of Distributed Computation in Wireless Sensor Networks," in *IEEE INFOCOM*, 2005.

[12] S. A. Hofmeyr, A. Somayaji, and S. Forrest, "Intrusion Detection using Sequences of System Calls," *Journal of Computer Security*, vol. 6, pp. 151–180, 1998.

[13] E. D. Manley, H. A. Nahas, and J. S. Deogun, "Localization and Tracking in Sensor Systems," in *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, June 2006.

[14] Z. Wang and J. Crowcroft, "Eliminating Periodic Packet Losses in the 4.3-Tahoe BSD TCP Congestion Control Algorithm," *ACM Computer Communication Review*, vol. 22, no. 2, pp. 9–16, April 1992.

[15] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized Gossip Algorithm," *IEEE/ACM Trans. Netw.*, vol. 14, pp. 2508–2530, June 2006.

[16] P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, March 2000.

[17] ——, "Critical Power for Asymptotic Connectivity in Wireless Networks," *Stochastic Analysis, Control, Optimization and Applications: A Volume in Honor of W.H. Fleming*, pp. 547–566, 1998.

[18] S. Kulkarni, A. Iyer, and C. Rosenberg, "An Address-light, Integrated MAC and Routing Protocol for Wireless Sensor Networks," *IEEE/ACM Trans. Netw.*, vol. 14, no. 4, pp. 793–806, 2006.

[19] Y. Zhou and M. Medidi, "Sleep-based Topology Control for Wakeup Scheduling in Wireless Sensor Networks," in *IEEE SECON*, June 2007.

[20] W. Pak, J.-G. Choi, and S. Bahk, "Tier Based Anycast to Achieve Maximum Lifetime by Duty Cycle Control in Wireless Sensor Networks," in *IWCMC*, August 2008.

Changhee Joo received his Ph.D degree from Seoul National University, Korea, 2005. He is an assistant professor at Ulsan National Institute of Science and Technology (UNIST), Korea. Before joining UNIST, he worked at Purdue University and the Ohio State University, USA, and at Korea University of Technology and Education, Korea. His research interests span a wide area of networking technologies including analysis, modelling, controls, and optimization. He has served on the technical committees of several primary conferences, including IEEE INFOCOM, ACM MobiHoc, and IEEE Globecom. He is a member of IEEE, and a recipient of the IEEE INFOCOM 2008 best paper award.

Ness B. Shroff received his Ph.D. degree in Electrical Engineering from Columbia University in 1994. He joined Purdue university immediately thereafter as an Assistant Professor in the school of ECE. At Purdue, he became Full Professor of ECE in 2003 and director of CWSA in 2004, a university-wide center on wireless systems and applications. In July 2007, he joined The Ohio State University, where he holds the Ohio Eminent Scholar endowed chair professorship in Networking and Communications, in the departments of ECE and CSE. From 2009-2012, he served as a Guest Chaired professor of Wireless Communications at Tsinghua University, Beijing, China, and currently holds an honorary Guest professor at Shanghai Jiaotong University in China.

His research interests span the areas of communication, social, and cyberphysical networks. He is especially interested in fundamental problems in the design, control, performance, pricing, and security of these networks. Dr. Shroff is a past editor for IEEE/ACM Trans. on Networking and the IEEE Communication Letters. He currently serves on the editorial board of the Computer Networks Journal, IEEE Network Magazine, and the Networking Science journal. He has chaired various conferences and workshops, and co-organized workshops for the NSF to chart the future of communication networks.

Dr. Shroff is a Fellow of the IEEE and an NSF CAREER awardee. He has received numerous best paper awards for his research, e.g., at IEEE INFOCOM 2008, IEEE INFOCOM 2006, Journal of Communication and Networking 2005, Computer Networks 2003 (also one of his papers was a runner-up at IEEE INFOCOM 2005), and also student best paper awards (from all papers whose first author is a student) at IEEE WiOPT 2012 and IEEE IWQoS 2006.