International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:10, No:6, 2016

Data Collection with Bounded-Sized Messages in Wireless Sensor Networks

Min Kyung An

***Abstract*—In this paper, we study the data collection problem in Wireless Sensor Networks (WSNs) adopting the two interference models: The graph model and the more realistic physical interference model known as Signal-to-Interference-Noise-Ratio (SINR). The main issue of the problem is to compute schedules with the minimum number of timeslots, that is, to compute the minimum latency schedules, such that data from every node can be collected without any collision or interference to a sink node. While existing works studied the problem with unit-sized and unbounded-sized message models, we investigate the problem with the bounded-sized message model, and introduce a constant factor approximation algorithm. To the best known of our knowledge, our result is the first result of the data collection problem with bounded-sized model in both interference models.

***Keywords*—Data collection, collision-free, interference-free, physical interference model, SINR, approximation, bounded-sized message model, wireless sensor networks, WSN.

I. Introduction

WSNs consist of a number of tiny wireless sensor devices (nodes). These nodes are scheduled to turn on their power to emit signals (i.e., to send data), or turn it off to conserve their limited power for specific time duration. When emitting signals, a *collision* or *interference* can occur at a node if the data transmission is interfered by signals concurrently sent by other nodes. In this case, the data should be re-transmitted. Because the tiny nodes have limited energy resources, it is crucial to reduce such unnecessary retransmissions in order to prolong the network's lifetime.

One important task of a WSN is to collect data periodically and send (forward) the data to a *sink node* in the network. This type of application is commonly known as *data collection*. An interesting approach for the data collection is to assign *timeslots* to nodes to obtain a good *schedule* through which data from every node is collected to the sink node. Here, if nodes are assigned the same timeslot in a schedule, then they can send data concurrently without causing any collision or interference. The objective of the problem is to compute schedules with the minimum number of timeslots, that is, to compute the minimum *latency* schedules, such that data from every node can be collected without any collision or interference.

For the data collection problem, there are three models in the literature: *Unit-sized*, *bounded-sized*, or *unbounded-sized* messages. In the unit-sized message model, a node can send a single unit-sized message at a timeslot, and therefore merging (combining) messages is not allowed. In the bounded-sized message model, a node can merge messages up to some limit before it sends, whereas in the unbounded-sized message model, there is no limit on the length of the merged message [1].

The data collection problem has been widely investigated by researchers in two interference models: The *graph model* and the *physical interference model*. In the graph model, given a *transmission range* $r(u)$ for every node u (i.e., the radius of the broadcasting disk covered by the signal sent by u using its transmission power $p(u)$), the *interference range* of u is defined as $\rho \cdot r(u)$, where $\rho \geq 1$ is the *interference factor* [2]. When $\rho = 1$, it is called a *collision-free graph model* that concerns *collision* only, and when $\rho \geq 1$, it is called a *collision-interference-free graph model* that concerns both collision and interference. Although the traditional graph model has been widely used in many studies, it is not an adequate model since *cumulative interference* caused by all the other concurrently transmitting nodes is ignored [2]. Thus, the more realistic *physical interference model* which is known as SINR has been used by many researchers for investigating problems in WSNs since its introduction by Gupta et al. in [3].

In the graph model, Bermond et al. [9] and Coleri et al. [23] proved the NP-hardness of the data collection problem when $\rho \geq 1$ and $\rho = 1$, respectively, with the unit-sized message model. With the unbounded-sized message model, the data collection is also known as *data aggregation*, and Chen et al. [24] and An et al. [22] proved the NP-hardness of the problem with $\rho = 1$ and $\rho \geq 1$, respectively. Because of the NP-hardness of the problem, many researchers have focused on proposing approximation algorithms, and the existing approximation algorithms with the unit-sized and unbounded-sized message models are summarized in Table I. Note that results in [4]–[6], [8]–[10] apply to special topologies or general graphs only. Lastly, with the bounded-sized message model, there currently exist no studies which investigated the data collection problem, to the best of our knowledge. There exist few studies [25], [26] which investigated a related application called *gossiping* assuming that messages can be merged into a single message whose size is bounded by $\log n$, where n is the number of nodes in a network.

In the SINR model, few researchers have investigated the data collection problem with the unit-sized message model, whereas there exists several studies which proposed approximation algorithms with the unbounded-sized message model, and Lam et al. [16], [17] showed the first result of the NP-hardness of the problem with the model. Like the

M. K. An is with Sam Houston State University, Huntsville, TX 77341, USA (e-mail: an@shsu.edu).

TABLE I
EXISTING APPROXIMATION ALGORITHMS FOR DATA COLLECTION

Message Model	Graph Model	Physical Interference Model
Unit-sized	[2], [4]–[6]: 3-approximation ($\rho = 1$)	[2], [7]: $O(1)$-approximation
	[8]: 2-approximation ($\rho = 1$)	
	[5]: 4-approximation ($\rho > 1$)	
	[2]: $O(1)$-approximation ($\rho > 1$)	
	[9]: 4-approximation ($\rho \geq 1$)	
	[6]: 4-approximation ($\rho = 2$)	
	[10]: $(1 + \frac{2}{\rho})$-approximation ($\rho \geq 2$)	
Bounded-sized	This paper: $O(1)$-approximation ($\rho \geq 1$)	This paper: $O(1)$-approximation
Unbounded-sized	[11]–[14]: $O(1)$-approximation ($\rho = 1$)	[15]–[20]: $O(1)$-approximation
	[21], [22]: $O(1)$-approximation ($\rho \geq 1$)	

graph model, there currently exists no studies investigating the problem with the bounded-sized message model. Existing approximation algorithms for the data collection in the SINR model are also summarized in Table I.

In this paper, we continue the study of the data collection problem in both the graph and SINR models. While existing works studied the problem with the unit-sized and unbounded-size message models only, we investigate the problem with the bounded-sized message model, and introduce a constant factor approximation algorithm which can be used not only for the graph model, but also for the SINR model.

This paper is organized as follows. Section II describes our network models and defines the data collection problem. In Section III, we introduce a constant factor approximation algorithm for the problem, and analyze it in Section IV. Finally, Section IV contains some concluding remarks.

II. PRELIMINARIES

A. Network Models

In this paper, we consider a WSN that consists of a set V of sensor nodes deployed in a plane. Each node $u \in V$ is assigned a transmission power level $p(u)$, and its *transmission range* $r(u)$ is defined as the radius of the broadcasting disk covered by the signal sent by u using its power $p(u)$. Accordingly, a directed edge (u, v) exists from node u to node v, if v resides in u's broadcasting disk (i.e., $d(u, v) \leq r(u)$, where $d(u, v)$ denotes the Euclidean distance between u and v.)

1) Graph Model: Let $C_u = \{v \,|\, v \in V,\, d(u, v) \leq r(u)\}$ denote the set of nodes that reside in u's transmission range. Then, two nodes u and v can communicate each other if $u \in C_v$ and $v \in C_u$. Next, let I_u denotes the set of nodes that reside in u's *interference range* $\rho \cdot r(u)$, where $\rho \geq 1$ is the *interference factor*. Then, the *collision* is said to occur at node w if there exist other concurrently sending nodes u and v such that $w \in C_u \cap I_v$, where $\rho = 1$ (i.e., $C_u = I_u$). Also, the *interference* is said to occur at node w if there exist other concurrently sending nodes u and v such that $w \in C_u \cap I_v$, where $\rho > 1$ (i.e., $C_u \subset I_u$).

In the graph model, we model a communication graph as a directed graph $G = (V, E)$ where $E = \{(u, v) \,|\, u, v \in V,\, d(u, v) \leq r(u)$ and $d(v, u) \leq r(v)\}$.

2) SINR Model: In the SINR model, when a node u sends data using its power level $p(u)$, the signal sent to a receiver v may not be strong enough to be received and hence the transmitted data is lost. It is because the signal sent by u fades and v is interfered by the cumulative interference caused by all the other concurrently transmitting nodes. In this model, the received power at the receiver v is defined as $p(u) \cdot d(u, v)^{-\alpha}$, where $\alpha > 2$ is the path loss exponent, and v can receive the data transmitted by the sender u without any interference only if the ratio of the received power at v to the total interference caused by all the other concurrently transmitting nodes and background noise is beyond an SINR threshold $\beta \geq 1$.

Formally, node v can successfully receive data via the communication edge (u, v) only if

$$SINR_{(u,v)} = \frac{\frac{p(u)}{d(u,v)^{\alpha}}}{N + \sum_{w \in X \setminus \{u,v\}} \frac{p(w)}{d(w,v)^{\alpha}}} \geq \beta \geq 1 \qquad (1)$$

where $N > 0$ is the background noise, and X is the set of other concurrently transmitting nodes.

As u can send its data to the nodes within the distance $(\frac{p(u)}{N\beta})^{\frac{1}{\alpha}}$ (i.e., $r(u) = (\frac{p(u)}{N\beta})^{\frac{1}{\alpha}}$) only, we model the communication graph as a directional disk graph $G = (V, E)$, where $E = \{(u, v) \,|\, u, v \in V,\, d(u, v) \leq (\frac{p(u)}{N\beta})^{\frac{1}{\alpha}}$ and $d(v, u) \leq (\frac{p(v)}{N\beta})^{\frac{1}{\alpha}}\}$, as in [2].

B. Problem Definition

We define the Minimum Latency Collection Scheduling (MLCS) problem as follows. Given a set of nodes for a network in a plane, we assign every node a *timeslot* such that nodes assigned the same timeslot, say t, can send data to their receivers simultaneously, satisfying the following conditions:

- (Graph Model) Neither collision nor interference occurs at any receiver.
- (SINR Model) The SINR inequality (1) is satisfied for every receiver.

A *schedule* is defined as a sequence of such timeslots, $(t_1, t_2, \cdots, t_L)$, where L denotes the *latency* of the schedule. A schedule is *successful* if all data of every node $v \in V - \{s\}$ is collected to a sink node $s \in V$. See Table II for notations.

III. CONSTANT FACTOR APPROXIMATION ALGORITHM

In this section, we introduce our constant factor approximation algorithm for the MLCS problem with the bounded-sized message model where each node can merge messages into a single message up to size of K before it sends. We further assume that every node u has its buffer storage $B(u)$ whose size is unlimited, and is assigned the transmission power level P, i.e., for every $u \in V$, $p(u) = P$.

TABLE II
NOTATIONS

Symbol	Definition
$r(u)$	Transmission range of node u
$p(u)$	Transmission power of node u
ρ	Interference factor
$\rho \cdot r(u)$	Interference range of node u, $\rho \geq 1$
C_u	The set of nodes that reside in $r(u)$
I_u	The set of nodes that reside in $\rho \cdot r(u)$
V	The set of nodes
E	The set of edges
G	A directed graph with V and E (Section II) A undirected graph with V and E (Section III)
(u, v)	A directional edge from u to v (Section II) A undirectional edge between u and v (Section III)
$d(u, v)$	The Euclidean distance between two nodes u and v
n	The number of nodes
α	Path loss exponent
β	SINR Threshold
N	Background noise
X	The set of concurrently sending nodes
$B(u)$	The buffer storage of node u
$m(u)$	The message of u
M	A merged message
K	The limit of the size of a combined message
T	A collection tree
$\ell(u)$	The level of u on T
h	The height of T
S_i	The set of sender nodes whose level is i on T, $1 \leq i \leq h$
$parent(v)$	A parent node of v on T
t	A timeslot
L	Latency (i.e., the length of schedule)
t_i	The i-th timeslot ($1 \leq i \leq L$)

A. Interference Models

We consider both graph and physical interference (SINR) models with the following assumptions as in [2]:

1) Graph Model: We set the maximum link length (i.e., the maximum transmission range) r to be the given P, and make the assumption that the undirected unit disk graph G, where $E = \{(u, v) \,|\, d(u, v) \leq r\}$, is connected and its interference factor $\rho \geq 1$.

2) SINR Model: From the SINR inequality (1) (Section II), we can compute the possible maximum link length as $r_{max} = (\frac{P}{N\beta})^{\frac{1}{\alpha}}$. We do not consider the links whose length is r_{max} because only node u can be a sender to send its data to some receiver v, where $d(u, v) = r_{max}$ (i.e., other nodes cannot transmit concurrently). Thus, we consider the links (u, v), where $d(u, v) \leq \delta(\frac{P}{N\beta})^{\frac{1}{\alpha}}$, for some constant $\delta \in (0, 1)$ as in [15] thereby setting r to be $\delta(\frac{P}{N\beta})^{\frac{1}{\alpha}}$. We also make the assumption that the undirected graph G, where $E = \{(u, v) \,|\, d(u, v) \leq r\}$, is connected and $\alpha > 2$ [3].

B. Algorithm

MLCS algorithm starts by constructing a *collection tree* T which is a breadth-first-search (BFS) tree (cf. [27]) on G rooted at the sink node s. Then, a number of iterations are performed to find a schedule based on T. Assigning timeslots for data collection is based on a constant value H. The value H guarantees that for any two sender nodes u and u's descendant node v on T, if $|\ell(u) - \ell(v)| \geq H$, then they can send data simultaneously without interference, where $\ell(u)$ denotes the level of u on the T. The constant value H is set as follows in the two interference models (See Lemmas 1 and 2):

- Graph Model: $H = \lceil \rho + 2 \rceil$
- SINR Model: $H = \lceil \{ (\frac{P \cdot 2\pi}{N(\delta^{-\alpha}-1)(\alpha-2)})^{\frac{1}{\alpha-2}} \} \cdot r^{-1} + 1 \rceil$

The details of data collection scheduling are contained in Algorithms 1 and 2.

Algorithm 1 MLCS

Input: A set V of nodes and a starting time slot t
Output: Length of schedule

1: Construct a collection tree T.
2: **for** each $u \in V - \{s\}$ **do**
3: $\quad \ell(u) \leftarrow$ level of u on T
4: $\quad B(u) \leftarrow \{m(u)\}$
5: $\quad S_i \leftarrow S_i \cup \{u\}$ where $i = \ell(u)$
6: **end for**
7: **repeat**
8: $\quad$ **for** $j = H$ downto 1 **do**
9: $\quad\quad \mathcal{S} \leftarrow \{S_i \,|\, i \,\%\, H = j \,\%\, H, 1 \leq i \leq h\}$
10: $\quad\quad t \leftarrow \mathbf{CS}(\mathcal{S}, t)$
11: $\quad$ **end for**
12: **until** $|B(u)| = 0$ for every $u \in V - \{s\}$
13: **return** $t - 1$

Once the collection tree T is constructed (Step 1 in Algorithm 1), $\ell(u)$ and $B(u)$ are initialized for every node $u \in V - \{s\}$. Then nodes are grouped by each level as S_1, $S_2, \cdots, S_h$, where h is the height of T (Steps 2-6 in Algorithm 1).

Algorithm 2 Collection Scheduling (**CS**)

Input: A set $\mathcal{S}$ and a starting time slot t
Output: Timeslot t

1: **for** each $S_i \in \mathcal{S}$ **do**
2: $\quad$ Pick one node $u \in S_i$ whose $|B(u)|$ is largest.
3: $\quad$ **if** $|B(u)| \neq 0$ **then**
4: $\quad\quad$ **if** $0 < |B(u)| \leq K$ **then**
5: $\quad\quad\quad$ Extract all the $|B(u)|$ messages from $B(u)$ and combine those message into a single message M.
6: $\quad\quad$ **else if** $|B(u)| > K$ **then**
7: $\quad\quad\quad$ Extract only $|B(u)|$ messages from $B(u)$ and combine those messages into a single message M.
8: $\quad\quad$ **end if**
9: $\quad\quad$ Assign timeslot t to u to send M to $parent(u)$.
10: $\quad\quad$ Store each message in M in $parent(u)$'s buffer $B(parent(u))$.
11: $\quad$ **end if**
12: **end for**
13: **return** $t + 1$

Next, each main iteration (Steps 7-12 in Algorithm 1) are repeated until the sink node s collects messages from all the other nodes, i.e., $|B(u)| = 0$ for every $u \in V - \{s\}$. In details, each main iteration repeats the following H submain iterations (Steps 8-11 in Algorithm 1):

- The first submain iteration examines $\mathcal{S} = \{S_H, S_{2H}, S_{3H}, S_{4H}, \cdots\}$
- The second submain iteration examines $\mathcal{S} = \{S_{H-1}, S_{2H-1}, S_{3H-1}, S_{4H-1}, \cdots\}$
- $\cdots$
- The H-th submain iteration examines $\mathcal{S} = \{S_{H-(H-1)}, S_{2H-(H-1)}, S_{3H-(H-1)}, S_{4H-(H-1)}, \cdots\}$

The details of each of the H submain iteration is as follows (Algorithm 2). For each of the sender sets in $\mathcal{S}$, only one node, say u, whose $|B(u)|$ is largest, is chosen. If there are less than or equal to K messages in its buffer, then extract all the $|B(u)|$ messages from the buffer, otherwise, if there are more than K messages in its buffer, then extract only $|B(u)|$ messages from the buffer. Then the extracted messages are merged into a single message M, and the node u is scheduled to forward M to its parent node $parent(u)$ on T at the timeslot t. At the timeslot t, M is unmerged and the unmerged messages are buffered at $parent(u)$'s buffer.

IV. Analysis

In this section, we analyze the Minimum Latency Collection Scheduling (MLCS) algorithm (Algorithm 1) and bound the latency of schedules produced by it.

First, we set the constant value H for the graph and SINR models.

Lemma 1 (Graph Model): For an interference factor $\rho \geq 1$, let $H = \lceil \rho + 2 \rceil$. In MLCS algorithm, for any two nodes u and u's descendant v on T, if $|\ell(u) - \ell(v)| \geq H$, then they can send data simultaneously without interference.

Proof: Consider a pair of sender and receiver, denoted by s_1 and r_1, and let s_2 be the *closest* sender to r_1 that does *not* interfere with r_1. Without loss of generality, let us assume r_1 is a descendant of s_1 and s_2 is a descendant of r_1 on T. Then, $d(r_1, s_2) > \rho \cdot r$. In order to bound the *shortest* number of hops between r_1 and s_2, assume a straight line between r_1 and s_2, and relay nodes with the power level P on the line. As we are assuming that r_1 and s_2 are *connected* with the shortest number of hops, we need at least $\lceil \frac{\rho \cdot r}{r} \rceil = \lceil \rho \rceil$ relay nodes for the connection. This implies that there are at least $\lceil \rho+1 \rceil$ hops between r_1 and s_2. Thus, in the MLCS algorithm, we can set $H = \lceil \rho + 2 \rceil$. ■

Lemma 2 (SINR Model): For SINR threshold $\beta \geq 1$, path loss exponent $\alpha > 2$, background noise $N > 0$, and some constant $\delta \in (0, 1)$, let $H = \lceil \tau \cdot r^{-1} + 1 \rceil$, where $\tau = (\frac{P \cdot 2\pi}{N(\delta^{-\alpha}-1)(\alpha-2)})^{\frac{1}{\alpha-2}}$ and $r = \delta(\frac{P}{N\beta})^{\frac{1}{\alpha}}$. In the MLCS algorithm, for any two nodes u and u's descendant v on T, if $|\ell(u) - \ell(v)| \geq H$, then they can send data simultaneously without interference.

Proof: Consider a sender s_1 trying to send its data to its farthest possible receiver r_1, and let s_2 be the *closest* sender to r_1 that does *not* interfere with r_1. Without loss of generality, let us assume r_1 is a descendant of s_1 and s_2 is a descendant of r_1 on T. Then $\tau = \left(\frac{P \cdot 2\pi}{N(\delta^{-\alpha}-1)(\alpha-2)}\right)^{\frac{1}{\alpha-2}}$ is a lower bound for the shortest distance between r_1 and s_2 [28], and therefore $d(r_1, s_2) \geq \tau$.

Next, let us bound the *shortest* number of hops between r_1 and s_2 as follows. Assume a straight line between r_1 and s_2, and relay nodes with the power level P on the line. As we are assuming that r_1 and s_2 are *connected* with the shortest number of hops, we need at least $\lceil \tau \cdot r^{-1} - 1 \rceil$ relay nodes for the connection. This implies that there are $\lceil \tau \cdot r^{-1} \rceil$ hops between r_1 and s_2. Thus, in the MLCS algorithm, we can set $H = \lceil \tau \cdot r^{-1} + 1 \rceil$. ■

Next, we bound the latency of the data collection schedules produced by the algorithm.

Lemma 3 (Lower Bound): If n is the number of nodes in a network, then every data collection schedule with bounded-sized model where several messages can be merged into a single message whose size is bounded by K takes at least $\lfloor \frac{n-1}{K} \rfloor$ timeslots.

Proof: Consider a node u, and $n-1$ messages that the node u has to receive. As a node can merge up to K messages, the node must receive at least $\lfloor \frac{n-1}{K} \rfloor$ distinct messages. Therefore, any data collection schedule allowed to merge messages up to size of K needs at least $\lfloor \frac{n-1}{K} \rfloor$ timeslots. ■

Theorem 4: The MLCS algorithm collects data from all the other nodes successfully to sink node s with at most $H \cdot \lceil \frac{n-1}{K} \rceil$ timeslots, and it is a constant-factor approximation with the factor of $2H$.

Proof: First note that there are $n-1$ messages that the sink node s must receive. In the MLCS algorithm, s receives single merged message every H timeslot, and as the subroutine, Collection-Scheduling algorithm (Algorithm 2), merges messages up to size of K, s receives at most $\lceil \frac{n-1}{K} \rceil$ messages to collect data without collision or interference (Lemmas 1 and 2). Therefore, it takes at most $H \cdot \lceil \frac{n-1}{K} \rceil$ timeslots.

Next, letting SOL denote the upper bound of the latency of the algorithm, and OPT be the lower bound (Lemma 3), we get $\frac{SOL}{OPT} \leq \frac{H \cdot \lceil \frac{n-1}{K} \rceil}{\lfloor \frac{n-1}{K} \rfloor} \leq 2H$. Thus, it is an approximation algorithm with the constant-factor of $2H$. ■

V. Conclusion

In this paper, we focused on the Minimum Latency Collection Scheduling (MLCS) problem of WSNs in the graph model as well as the more realistic physical interference model known as SINR. We proposed a $O(1)$-approximation algorithm that works in both the interference models with bounded-sized message model. To the best known of our knowledge, our result is the first result of the problem with bounded-sized model in both interference models. For future work, we plan to study another related problem, gossiping, adopting both the interference models with bounded-sized message model.

References

[1] M. K. An, N. X. Lam, D. T. Huynh, and T. N. Nguyen, "Minimum Latency Gossiping in Wireless Sensor Networks," in *Proceedings of the International Conference on Wireless Networks (ICWN)*, 2012, pp. 406–412.

[2] M. K. An and H. Cho, "Efficient Data Collection in Interference-Aware Wireless Sensor Networks," *Journal of Networks (JNW)*, 2015, to appear.

[3] P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks," *IEEE Trans. on Information Theory*, vol. 46, pp. 388–404, 2000.

[4] C. Florens and R. J. McEliece, "Packet Distribution Algorithms for Sensor Networks," in *INFOCOM*, 2003.

[5] V. Bonifaci, P. Korteweg, A. Marchetti-Spaccamela, and L. Stougie, "An Approximation Algorithm for the Wireless Gathering Problem," in *SWAT*, 2006, pp. 328–338.

[6] J.-C. Bermond, B. Li, N. Nisse, H. Rivano, and M.-L. Yu, "Data Gathering and Personalized Broadcasting in Radio Grids with Interferences," INRIA, Research Report RR-8218, 2013.

[7] P.-J. Wan, L. Wang, and O. Frieder, "Fast Group Communications in Multihop Wireless Networks Subject to Physical Interference," in *IEEE 6th Intern. Conf. on Mobile Adhoc and Sensor Systems (MASS)*, 2009, pp. 526–533.

[8] D. R. Kowalski, E. Nussbaum, M. Segal, and V. Milyeykovski, "Scheduling Problems in Transportation Networks of Line Topology," *Optimization Letters*, vol. 8, no. 2, pp. 777–799, 2014.
[9] J.-C. Bermond, J. Galtier, R. Klasing, N. Morales, and S. Perennes, "Hardness and Approximation of Gathering in Static Radio Networks," *Parallel Processing Letters*, vol. 16, no. 2, pp. 165–184, 2006.
[10] J.-C. Bermond, L. Gargano, S. Prennes, A. A. Rescigno, and U. Vaccaro, "Optimal Time Data Gathering in Wireless Networks with Multidirectional Antennas," *Theoretical Computer Science*, vol. 509, pp. 122–139, 2013.
[11] S. C. H. Huang, P.-J. Wan, C. T. Vu, Y. Li, and F. Yao, "Nearly Constant Approximation for Data Aggregation Scheduling," in *INFOCOM*, 2007, pp. 6–12.
[12] B. Yu, J. Li, and Y. Li, "Distributed Data Aggregation Scheduling in Wireless Sensor Networks," in *INFOCOM*, 2009, pp. 2159–2167.
[13] X. Xu, S. Wang, X. Mao, S. Tang, P. Xu, and X.-Y. Li, "Efficient Data Aggregation in Multi-Hop WSNs," in *GLOBECOM*, 2009, pp. 3916–3921.
[14] X. Xu, X. Y. Li, X. Mao, S. Tang, and S. Wang, "A Delay-Efficient Algorithm for Data Aggregation in Multihop Wireless Sensor Networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, pp. 163–175, 2011.
[15] X.-Y. Li, X. Xu, S. Wang, S. Tang, G. Dai, J. Zhao, and Y. Qi, "Efficient Data Aggregation in Multi-hop Wireless Sensor Networks under Physical Interference Model," in *MASS*, 2009, pp. 353–362.
[16] N. X. Lam, M. K. An, D. T. Huynh, and T. N. Nguyen, "Minimum Latency Data Aggregation in the Physical Interference Model," in *MSWiM*, 2011, pp. 93–101.
[17] M. K. An, N. X. Lam, D. T. Huynh, and T. N. Nguyen, "Minimum Latency Data Aggregationg in the Physical Interference Model," *Computer Communications*, vol. 35, no. 18, pp. 2175–2186, 2012.
[18] M. K. An, N. X. Lam, D. T. Huynh, and T. N. Nguyen, "Minimum Latency Aggregation Scheduling in Interference-Aware 3-Dimensional WSNs," in *International Conference on Wireless Networks (ICWN)*, 2013.
[19] N. X. Lam, M. K. An, D. T. Huynh, and T. N. Nguyen, "Scheduling Problems in Interference-Aware Wireless Sensor Networks," in *ICNC*, 2013, pp. 783–789.
[20] N. X. Lam, T. Tran, M. K. An, and D. T. Huynh, "A Note on the Complexity of Minimum Latency Data Aggregation Scheduling with Uniform Power in Physical Interference Model," *Theoretical Computer Science*, vol. 569, pp. 70–73, 2015.
[21] P. Wan, S. C. Huang, L. Wang, Z. Wan, and X. Jia, "Minimum-Latency Aggregation Scheduling in Multihop Wireless Networks," in *MOBIHOC*, 2009, pp. 185–194.
[22] M. K. An, N. X. Lam, D. T. Huynh, and T. N. Nguyen, "Minimum Data Aggregation Schedule in Wireless Sensor Networks," *International Journal of Computers and Their Applications (IJCA)*, vol. 18, no. 4, pp. 254–262, 2011.
[23] S. C. Ergen and P. Varaiya, "TDMA Scheduling Algorithms for Wireless Sensor Networks," *Wireless Networks*, vol. 16, no. 4, pp. 985–997, 2010.
[24] X. Chen, X. Hu, and J. Zhu, "Minimum Data Aggregation Time Problem in Wireless Sensor Networks," in *MSN*, 2005, pp. 133–142.
[25] R. Bar-Yehuda, A. Israeli, and A. Itai, "Multiple Communication in Multi-Hop Radio Networks," *SIAM Journal on Computing*, vol. 22, pp. 875–887, 1993.
[26] M. Christersson, L. Gasieniec, and A. Lingas, "Gossiping with Bounded Size Messages in Ad Hoc Radio Networks," in *ICALP*, 2002, pp. 377–389.
[27] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. MIT Press and McGraw-Hill, 2009.
[28] M. K. An, N. X. Lam, D. T. Huynh, and T. N. Nguyen, "Connectivity in Wireless Sensor Networks in the SINR Model," in *MASCOTS*, 2012, pp. 145–152.