An Algorithm for Data Aggregation Scheduling with Long-lifetime and Low-latency in Wireless Sensor Networks

Zhengyu Chen[1,2,3], Geng Yang[1,2], Lei Chen[1,2] and Jin Wang[4]

[1] *Key Laboratory of Broad band Wireless Communication & Sensor Networks Technology of Ministry of Education, Nanjing University of Posts &Telecommunications, Nanjing, China*

[2] *College of Computer Science & Technology, Nanjing University of Posts &Telecommunications, Nanjing, China*

[3] *School of Information Technology, Jinling Institute of Technology, Nanjing, China*

[4] *School of computer & software, Nanjing University of information science & technology, Nanjing 210044, China*

zych@jit.edu.cn, {yangg, chenlei }@njupt.edu.cn, wangjin@nuist.edu.cn

Abstract

In scenarios of real-time data collection in long-term deployed Wireless Sensor Networks (WSNs), low-latency data collection and long network lifetime become key issue. We propose a Long-Lifetime and Low-Latency Data Aggregation Scheduling algorithm (L^4DAS) in wireless sensor networks. Firstly, we formally formulate the problem of long-lifetime and minimum-latency aggregation scheduling as a constrained optimization problem, and then propose an approximation algorithm for this problem by constructing a degree-bounded minimum height spanning tree as aggregation tree and designing a maximum interference priority scheduling scheme to schedule the transmission of nodes in aggregation tree. Finally, through the simulation and comparisons, we prove the effectiveness of the algorithm.

Keywords: *WSNs, Data aggregation scheduling, Network lifetime, Long-lifetime, Low-latency*

1. Introduction

Wireless Sensor Networks have been used for many long-term and real-time applications which require networks to operate long durations, as well as to transmit the sensed data to sink as soon as possible. Therefore, both maximizing lifetime and minimizing delay are the fundamental requirements. However, these two requirements are usually conflict with the limited battery power and communication bandwidth of sensor node. Sleep-wake scheduling [1, 2] and data aggregation [3] are the effective mechanisms to prolong the lifetime of energy-constrained sensor networks. However, both sleep-wake scheduling and data aggregation can also lead to additional data collection delay [2, 3]. So, it is critical to research a problem of data aggregation scheduling with long-lifetime and low-latency in WSNs.

In WSNs, the Minimum-Latency Aggregation Schedule (MLAS) problem is to find a collision-free transmission schedule of data aggregation for all sensors such that the total time latency for aggregated data to reach the sink is minimized. Chen, *et al.*, in [6] proved that the problem of minimizing the latency of data aggregation is NP-hard. Many scheduling algorithms have been proposed to reduce the latency of data aggregation by using TDMA technology [4, 5, 6, 8]. Chen, *et al.*, in [4] designed an algorithm named SDA based on Shortest Path Tree with a latency bound of $(\Delta-1)R$, where Δ is the maximum degree and R is the network radius. Huang, *et al.*, designed an algorithm in [5] based on maximal independent sets which has an latency bound of $23R+\Delta-18$. Wan, *et al.*, [6] proposed three algorithms: SAS, PAS and EPAS which have latency bounds of $15R+\Delta-4$, $2R+O(\log R)+\Delta$ and $(1+O(\log R/\sqrt[3]{R}))R+\Delta$ respectively. Yu, *et al.*, [7] and Xu, *et al.*, [8] proposed a distributed scheduling method generating collision-free schedules with delay at most $24D+6\Delta+16$ and $16R+\Delta-14$ time-slots respectively, where D and R is the network diameter and radius. These algorithms which do not consider lifetime scheduling problem theoretically prove the upper-bound on latencies, but this upper bound is far too pessimistic compared to the typical practical behavior of their algorithm. Malhotra, *et al.*, [9] proved that a lower bound on the schedule length for a given tree is $\max\{|\xi_i|+d_i : i=1,2,...,N\}$, where ξ_i and d_i are the number of children and hop distance from the sink, respectively, for node i. Their scheme got the better performance than the previous algorithms by constructing a balanced shortest path tree (BSPT) and using a ranking-based heuristic scheduling.

Energy efficiency is the biggest challenge in designing long-living sensor networks. Wu, *et al.*, [10] proved that finding a maximum lifetime arbitrary tree is NP-complete, and proposed an approximation algorithm that produces a sub-optimal tree. Luo, *et al.*, [11] studied the problem of maximizing the lifetime for the shortest path aggregation trees by reducing it to a general version of semi-matching problem. Malhotra, *et al.*, [9] constructed a BSPT to prolong network lifetime which balance the number of children per node by assigning nodes from level h+1 to the parents at level h such that every parent has an equal number of children. However, the network lifetime in [9] determined by network architecture and the number of nodes, and the non-leafy characteristic of BSPT is not conducive to reducing scheduling length.

In this paper, we propose a Long-Lifetime and Low-Latency Data Aggregation Scheduling (L^4DAS) algorithm in WSNs. Our main contributions are as follows: (1) we construct a degree-bounded minimum height spanning tree as aggregation tree which provides a long network lifetime and is conducive to reduce scheduling length and, (2) we propose a maximum interference priority scheduling algorithm to schedule the transmission of nodes such that the latency is approximately minimized, and (3) we carry out extensive simulations to verify our algorithms, and the results show that our algorithm greatly outperforms the state-of-art schemes.

The rest of the paper is organized as follows. Section 2 describes system models, assumptions, and problem formulation. We present an algorithm for construction data aggregation tree with efficient lifetime and minimum radius in Section 3. A greedy maximum TDMA scheduling algorithm is also proposed in Section 3. Section 4 presents the simulation results and analysis. Section 5 concludes the paper.

2. System Model and Problem Statement

2.1 System Model and Definitions

We consider a WSN consisting of N sensor nodes $v_1, v_2, ..., v_N$ and sink node v_s. We assume that all nodes have the same transmission range r and interference range r_I. We use a undirected graph $G(V, E)$ to represent this WSN, where $V = \{s, v_1, v_2, ..., v_N\}$ denotes the set of nodes and E denotes the set of edges, *i.e.* there is an edge $(v_i, v_j) \in E$ whenever their Euclidean distance $\|v_i - v_j\| \le r$.

We consider the protocol interference model in which concurrent transmissions on two edges $u \to v$ and $p \to q$ conflict with each other if and only if $v = q$, $\|p - v\| \le r_I$ or $\|q - u\| \le r_I$ [7, 8, 12]. All sensors are homogeneous which have the same energy E_{node} and consume energy E_{tx} and E_{rx} for transmitting and receiving one bit data respectively. We adopt a perfect data aggregation model and TDMA-based scheduling protocol which offer the advantage of permitting nodes to enter into sleep mode during inactive periods so as to efficiently eliminate collisions and prevent overhearing. In order to facilitate the description, we give the following definitions:

Definition 1. Link Conflict. Under the protocol interference model, there exists a **Link Conflict** between link $u \to v$ and $p \to q$ if and only if $v = q$, $\|p - v\| \le r_I$ or $\|q - u\| \le r_I$.

Definition 2. **Node Conflict**. If there exists a link conflict between link $u \to v$ and $p \to q$, we call sender u and p are Node Conflict, or call that sender u conflict with sender p.

Definition 3. **Round.** A Round is defined as the process of gathering data from all nodes to the sink which is equivalent to a TDMA schedule period consisting of T time slots. The duration of each round is called the scheduling latency. At each time slot, all senders and their corresponding parent nodes are scheduled in active state while the remaining nodes in sleep state.

Definition 4. **Network Lifetime.** The network lifetime is defined as the lifetime of the first dead node in the network. In data aggregation scheduling, we usually need to construct a data aggregation tree, so the network lifetime will be the lifetime of data aggregation tree. The lifetime $L(T_{DA})$ of the data aggregation tree T_{DA} is

$$L(T_{DA}) = \min_{i=1,2,...N} \left\lfloor \frac{E_{node}}{KE_{rx}D(T_{DA}, v_i) + K(E_{tx} - E_{rx})} \right\rfloor \qquad \textbf{(1)}$$

where $D(T_{DA}, v_i)$ is the degree of node v_i in T_{DA}, K is the number of bits generated by each node in a round.

2.2 Problem Statement

We formulate the long-lifetime and low-latency data aggregation scheduling problem as a constrained optimization problem. Table 1 lists the symbols used in this optimization problem.

Table 1. Symbols and Meanings

Symbols	Meanings
N_{v_i}	The set denotes the neighbors of node v_i.
$f^t_{(v_i,v_j)}$	The decision variable of link (v_i, v_j) at time slot t; $f^t_{(v_i,v_j)} = 1$ if link (v_i, v_j) is scheduled to transmit at time slot t; $f^t_{(v_i,v_j)} = 0$ otherwise.
$R(v_i)$	The recipient corresponds to the sender v_i.
$t(v_i)$	The time slot is assigned for the transmission by node v_i.
$CHD(v_i)$	The set denotes the children nodes of v_i in aggregation tree.
D	The maximum permitted degree of each node in aggregation tree.
T	Scheduling latency

The objective function of this problem becomes Minimize T, subjects to the following constraints,

$$
\begin{aligned}
&\text{(a)} \sum_{v_k \in N_{v_i}} \sum_{t=1}^{T} f^t_{(v_i,v_k)} = 1 \qquad i \in \{1,2,...,N\} \\
&\text{(b)} \sum_{v_k \in N_{v_i}} \sum_{t=t(v_i)}^{T} f^t_{(v_k,v_i)} = 0 \qquad i \in \{1,2,...,N\} \\
&\text{(c)} \sum_{v_j \in N_{R(v_i)}} f^{t(v_i)}_{(R(v_i),v_j)} = 0 \qquad i \in \{1,2,...,N\} \qquad\qquad (2) \\
&\text{(d)} \sum_{v_j \in N_{R(v_i)}, v_j \neq v_i} \sum_{v_k \in N_{v_j}} f^{t(v_i)}_{(v_j,v_k)} + \sum_{v_q \in N_{v_i}, v_q \neq R(v_i)} \sum_{v_p \in N_{v_q}, v_p \neq v_i} f^{t(v_i)}_{(v_p,v_q)} = 0 \qquad i \in \{1,2,...,N\} \\
&\text{(e)} \sum_{v_k \in CHD(v_i)} \sum_{t=1}^{T} f^t_{(v_k,v_i)} \leq D-1 \qquad i \in \{1,2,...,N\}
\end{aligned}
$$

Constraint (a) enforces a single transmission per node in each round. Constraint (b) ensures that, once node v_i transmits, it can no longer receive data from its children nodes in the same round. As shown in Figure 1, node v_a can transmit data only after it has received data from all its children. Constraint (c) guarantees that node can not transmit and receive simultaneously at the same time slot, i.e. half-duplex operation. In Figure 1, $R(v_i)$ can not transmit data to v_b when v_i is transmitting data to $R(v_i)$. Constraint (d) ensures requirement that there can be no interference at the recipient node, *i.e.* in Figure 1, when a link $(v_i, R(v_i))$ is being scheduled at some time slot, link (v_i, v_k) and link (v_p, v_q) can not be scheduled at the same time slot. Constraint (e) guarantees that each node at most has $D-1$ children nodes. This scheduling problem has been proved as a NP-hard problem [4]. We propose an approximation algorithm for this problem. We first construct a degree-bounded minimum height spanning tree as aggregation tree, and then design a maximum interference priority scheduling scheme to schedule the transmission of nodes in aggregation tree. The details will be presented in the following section.

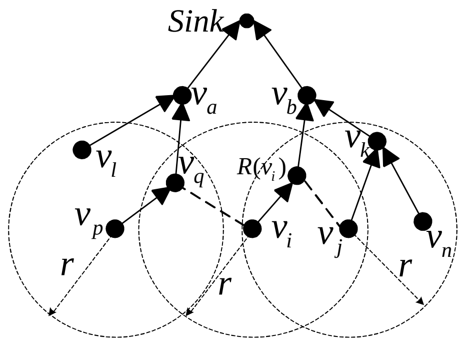


Figure 1. An Example of Schedule

3. Main Design

3.1 Data Aggregation Tree Construction

We construct the data aggregation tree through two phases. First, we break up the graph into clusters with the diameter equaling to transmission range r and construct a cluster spanning tree respecting the degree constraint D in each cluster. Then we construct a global tree over the clusters and connect the spanning tree in each cluster to the global tree.

Given $G(V,E)$, depending on the size of the deployment area and transmission radius, we first partition V into pairwise disjoint sets [13, 14]:

(1) $V = V_1 \bigcup V_2 \bigcup \ldots \bigcup V_m$, for $\forall i, j \in \{1,2,\ldots,m\}$, $V_i \bigcap V_j = \varnothing$;

(2) $\forall v_p, v_q \in V_i$, $i \in \{1,2,\ldots,m\}$, $\|v_p, v_q\| \le r$;

where V_i ($i \in \{1,2,\ldots,m\}$) is the set of nodes in cluster i. We partition V by tessellating the deployment area into a set of hexagonal clusters each of side length $r/2$ and assigning each node to a unique cluster whose center is closest to the node. We then choose a representative for each cluster to constitute a set $R = \{u_1, u_2, \ldots, u_m\}$ where $u_i \in V_i$, $i \in \{1,2,\ldots,m\}$. The global tree will be constructed by these representatives. If some representatives can not connect with each other, we choose some connecting nodes $C = \{c_1, c_2, \ldots, c_n\}$ to connect them. The global tree should be constructed satisfying the following constraint,

$$\begin{aligned} &\min_{R\in V} \max_{u\in R} \text{hops}(u, v_s) \\ s.t. \quad &\max_{u\in R} \deg(u) \le D-1 \end{aligned} \qquad (3)$$

where $\text{hops}(u, v_s)$ is the hop distance from u to sink v_s, $\deg(u)$ is the degree of u. As shown in Figure 2**,** each circle represents a cluster. The nodes from v_1 to v_6 are representatives which correspond to 6 clusters respectively. c_1 and c_2 are connecting nodes. Dotted lines with arrows connect these nodes together to form the global tree. In each cluster, we construct a cluster spanning tree rooted at representative node while respecting the degree constraint D. The Algorithm of constructing data aggregation tree is shown in **Algorithm 1**.

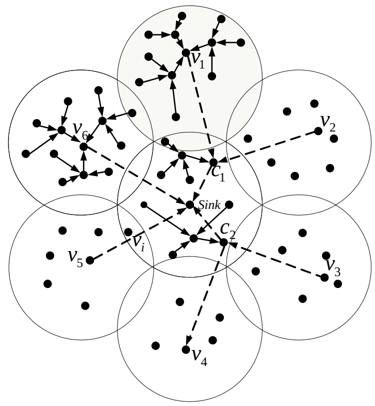


Figure 2. An Example of Constructing a Data Aggregation Tree

Algorithm 1 Constructing Data Aggregation Tree

Input: $G(V,E)$, sink, $D \geq 2$;

Output: aggregation tree T_{DA} .

Step 1: Partition V into pairwise disjoint sets $V_1, V_2, \ldots, V_m$.

Step 2: Choose representatives $R = \{u_1, u_2, \ldots, u_m\}$ and connecting nodes $C = \{c_1, c_2, \ldots, c_n\}$ satisfying formula (3);

Step 3: Construct global tree T_G rooted at sink by connecting nodes in both R and C ;

Run Step 4 to Step 6 for each cluster, construct m **cluster spanning trees**;

Step 4: For cluster i , $V_i = \{v_{i1}, v_{i2}, \ldots, v_{ij}\}$ and representative node u_i , initialize spanning tree $T_i = (V_{Ti}, E_{Ti})$, $V_{Ti} \leftarrow \{u_i\}$ $E_{Ti} \leftarrow \varnothing$;

Step 5: Choose n nodes ($\{v_{i1},...,v_{in}\}$) from V_i as children nodes of u_i , where $n = D - \deg^{T_G}(u_i)$ ($\deg^{T_G}(u_i)$ is the degree of node u_i in the global tree T_G), i.e. $V_{Ti} \leftarrow V_{Ti} \bigcup \{v_{i1},...,v_{in}\}$, $E_{Ti} \leftarrow E_{Ti} \bigcup \{(v_{i1},u_i),...,(v_{in},u_i)\}$;

Step 6: Choose nodes from $\{v_{i1},...,v_{in}\}$ as parent nodes, then choosing $D-1$ children nodes from $V_i / \{v_{i1},...,v_{in}\}$ for each parent node, repeat this process until all nodes in $V_i = \{v_{i1}, v_{i2}, \ldots, v_{ij}\}$ are scheduled;

Step 7: Combine global tree T_G with all cluster spanning trees, $T_{DA} = T_B \bigcap_{i=1}^{m} T_i$.

3.2 Aggregation Scheduling

A data aggregation schedule with delay T can be defined as a sequence of sender sets $S_1, S_2, ..., S_T$ satisfying the following conditions:

(1) $S_i \bigcap S_j = \varnothing, \ \forall i \neq j;$

(2) $\forall v_i, v_j \in S_l$, $l = 1, 2, ..., T$, v_i and v_j do not conflict with each other；

(3) At time slot k , each sender in S_k transmits data to its parent node in $V - \bigcup_{j=1}^{k} S_j$

In this subsection, we design a maximum interference priority scheduling scheme to schedule the nodes in aggregation tree. We give some definitions as follows:

Definition 5. **Eligible Schedule Set**. At a time slot, nodes are eligible to be scheduled as senders if they are leaf-nodes not being scheduled at earlier time slot or intermediate nodes whose all children nodes have been scheduled at earlier slots. We define a set F of such nodes as an eligible schedule set.

Definition 6. **Interference Intensity**. When a node is scheduled to transmit, the number of receivers which can be interfered by this node is defined as the interference intensity of this node. For an eligible schedule set F_i and the sets $S_1, S_2, ..., S_{i-1}$ that has been scheduled at earlier $i-1$ slots, we can calculate the interference intensity $DI(u)$ of node u (for $\forall u \in F_i$) as

$$DI(u) = \left| Nei_u \bigcap (V - \bigcup_{j=1}^{i-1} S_j) - Nei_u \bigcap F_i - \{par_u\} \right| \quad (4)$$

where Nei_u is the neighbors of node u in graph G, par_u is the parent node of node u in aggregation tree, and $V - \bigcup_{j=1}^{i-1} S_j$ represents the set of nodes that have not been scheduled at time slot i.

For the eligible schedule set F_i at time slot i, we first assign the node with the largest interference intensity to sender set S_i, then choose the node that has the largest interference intensity in $F_i - S_i$ and also do not conflict with the nodes in S_i. We continue this process until there do not exist node in $F_i - S_i$ satisfying conflict-free schedule. **Algorithm 2** gives the processes of schedule.

Algorithm 2 Aggregation Scheduling

Input: $G = (V, E)$, data aggregation tree T_{DA}, sink;

Output: Sets of sender $S_1, S_2, ..., S_T$;

Initialize: $t = 1$, $i = 1$;

Repeat executing Step 1 to Step 4 until all nodes have been scheduled.

Step 1：Calculate the eligible schedule set F_i depending on Definition 5 **at time slot** i;

Step 2: $v = \arg\max_{u \in F_i} DI(u)$, $S_i = \{v\}$;

Step 3: if $p = \arg\max_{u \in (F_i - S_i)} DI(u)$ and node p do not conflict with the nodes in S_i, $S_i \leftarrow \{p\}$; repeat executing **Step 3** until there do not exist node in $F_i - S_i$ satisfying conflict-free schedule;

Step 4: Output S_i, $i = i + 1$

4. Simulation Results

We evaluate the performance of our algorithm using simulations. We randomly deploy sensor nodes in a $200\text{m} \times 200\text{m}$ field with a sink located at (100, 100). All sensor nodes have the same transmission range and interference radius, *i.e.*, $r = r_I$. We generate 30 random networks and present the averaged results for performing comparisons. Table 2 shows the simulation parameters.

Table 2. Simulation Parameters

Parameters	Value
Number of nodes N	400~1000
Transmission range r	25, 30, 35.
Length of data	5 Bytes
Energy consumption for transmission E_{tx}	0.6 nJ/bit
Energy consumption for reception E_{rx}	0.2 nJ/bit
Energy of each node E_{node}	3 $J/node$
Degree constraint D in L^4DAS	4

For aggregation latency, we compare L^4DAS with WIRES [9] and LDAS [12]. Transmission range r is fixed to 25m, the number of nodes varies from 400 to 1000 with an increment of 50. As can be seen from Figure 3, both L^4DAS and LDAS outperform WIRES. When the number of nodes is less than 650, LDAS outperforms L^4DAS. The reason is that when node density is not high, the aggregation tree of L^4DAS has higher height that can increase the low-bound of latency [9]. However, with the increasing of node density, the degree of nodes (especially sink) in LDAS increases rapidly which causes the low-bound of latency increases. So, when the node density exceeds a certain value, L^4DAS will always outperform LDAS.

In Figure 4, we give the latency of L^4DAS with different r . As transmission range increases, the potential for conflict between the nodes increases, which leading to increase latency. In Figure 5, N varies from 500 to 1000 with an increment of 50, while r takes three values 25, 30 and 35 respectively. It is indicated from the histogram that with the increment of the number of nodes and the transmission range, the improvement of our algorithm will be larger. These results indicate that our algorithm is greatly preferred for large scale and high density WSNs.

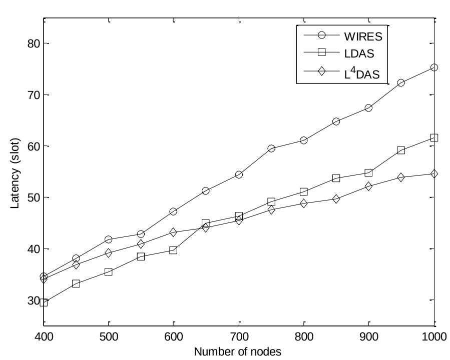


Figure 3. Latency with Different Number of Nodes

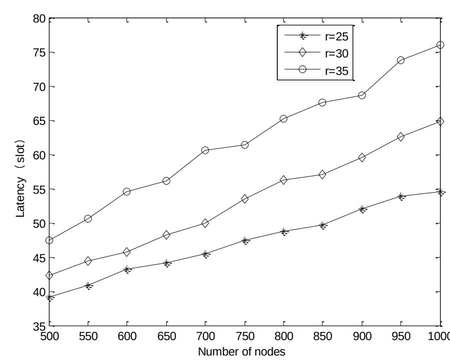


Figure 4. Latency of L^4DAS with Different r

Using the same simulation parameters as Figure 3, Figure 6 shows the network lifetime changes with the different number of nodes. It can be seen that our algorithm outperforms LDAS and WIRES. With the changes in the number of nodes, the network lifetime of our algorithm changes little while WIRES fluctuates larger.

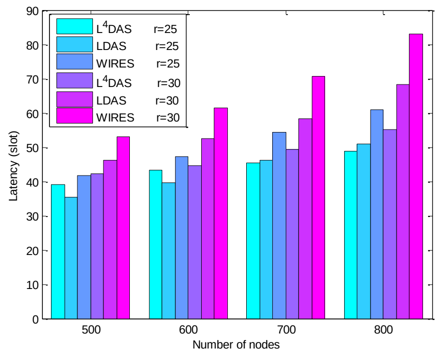


Figure 5. Latency with Different Number Number of Nodes and Transmission Range

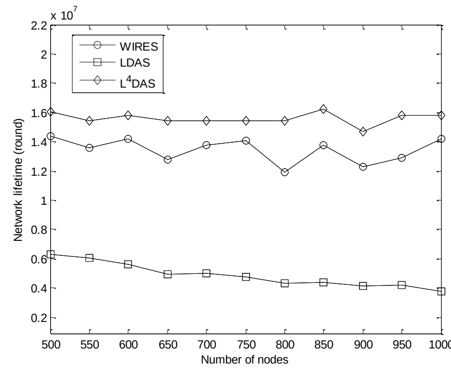


Figure 6. Network Lifetime with Different of Nodes

Figure 7 shows the data aggregation trees with 500 nodes of three algorithms. The leaf nodes of our algorithm and LDAS are more evenly distributed in the deployment area which is prone to schedule more nodes for parallel transmission in a time slot. The degrees on our algorithm and WIRES are more evenly distributed so as to provide longer lifetime than LDAS. However, due to being impacted by the network topology and node distribution, the degree of some nodes in WIRES is very high.

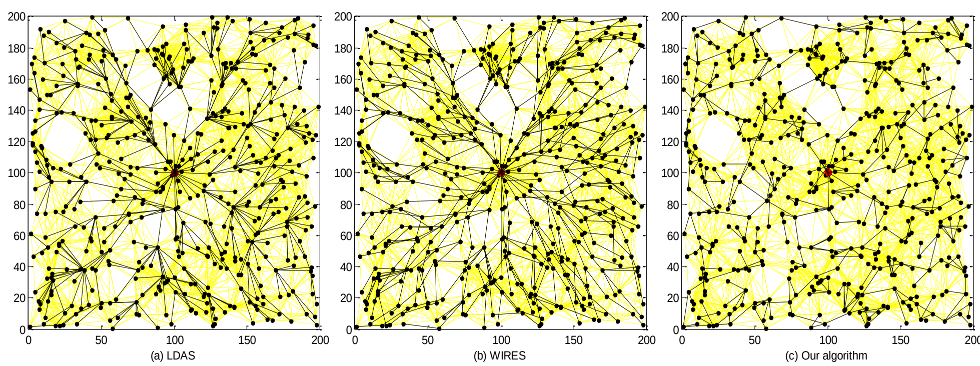


Figure 7. The Aggregation Tree of Different Algorithm

5. Conclusions and the Future Works

In this paper, we have investigated the data aggregation problem and considered its latency and network lifetime for WSNs in scenarios of real-time and long-term applications. We formulated the problem as a constrained optimization problem. Then, we proposed an approximation algorithm for this problem based on constructing a data aggregation tree and designing a maximum interference priority scheduling scheme. Finally, through the simulation and comparisons, we proved that our algorithm outperforms the start-of-art schemes. In the future, we will extend our scheduling algorithm for the more realistic physical interference model. We will also research the distributed algorithms for constructing aggregation tree and scheduling the transmission of nodes.

Acknowledgments

This work was supported by the National Basic Research Program of China (973 Program) under Grant No. 2011CB302903, the National Science Foundation of China under Grant No. 60873231, the Natural Science Foundation of Jiangsu Province under Grant No.BK2009426, the Innovation Project for postgraduate cultivation of Jiangsu Province under Grant No. CXZZ11_0402, CX10B_195Z, CXLX11_0415, CXLX11_0416, the Natural Science Research Project of Jiangsu Education Department under Grant No. 09KJD510008, the Natural Science Foundation of the Jiangsu Higher Education Institutions of China under Grant No. 11KJA520002. This work was also supported by Industrial Strategic Technology Development Program (10041740) funded by the Ministry of Knowledge Economy (MKE）Korea, and by the Natural Science Foundation of Jiangsu Province (No. BK2012461).

References

[1] T. V. Dam and K. Langendoen, “An adaptive energy-efficient MAC protocol for wireless sensor networks”, In the 1st International Conference on Embedded Networked Sensor Systems SenSys, ACM Press, Los Angeles, California, USA **(2003)**, pp. 171-180.

[2] J. Kim, X. Lin, N. B. Shroff and P. Sinha, “Minimizing Delay and Maximizing Lifetime for Wireless Sensor Networks With Anycast”, IEEE/ACM Transaction on Networking, vol. 18, no. 2 **(2010)**.

[3] B. Krishnamachari, D. Estrin and S. Wicker, “The Impact of Data Aggregation in Wireless Sensor Networks”, In the 22nd International Conference on Distributed Computing Systems Workshops, IEEE Press, Vienna, Austria **(2002)**, pp. 575-78.

[4] X. J. Chen, X. D. Hu and J. M. Zhu, “Minimum Data Aggregation Time Problem in Wireless Sensor Networks”, Lecture Notes in Computer Sciences, vol. 3794, **(2005)**, pp. 133-142.

[5] S. C. Huang, P. J. Wan, C. T. Vu, Y. S. Li and F. Yao, “Nearly Constant Approximation for Data Aggregation Scheduling in Wireless Sensor Networks”, In: the 26th Conference on Computer Communications, INFOCOM 2007, IEEE Press, Anchorage, Alaska, USA **(2007)**, pp. 366-372.

[6] P. J. Wan, S. C. Huang, L. X. Wang, Z. Y. Wan and X. H. Jia, “Minimum-latency Aggregation Scheduling in Multihop Wireless Networks”, In the 10th ACM International Symposium on Mobile Ad Hoc Networking and Computing, ACM Press, New Orleans, LA, USA **(2009)**, pp. 185-194.

[7] B. Yu, J. Z. Li and Y. S. Li, “Distributed Data Aggregation Scheduling in Wireless Sensor Networks”, In: the 28th Conference on Computer Communications, INFOCOM 2009, IEEE Press, Rio de Janeiro, Brazil **(2009)**, pp. 2159-2167.

[8] X. H. Xu, X. Y. Li, X. F.Mao, S. J. Tang and S. G. Wang, “A Delay-Efficient Algorithm for Data Aggregation in Multi-hop Wireless Sensor Networks”, IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 1, **(2011)**.

[9] B. Malhotra, I. Nikolaidis and M. A. Nascimento, “Aggregation Convergecast Scheduling in Wireless Sensor Networks”, Wireless Networks, vol. 17, no. 2, **(2011)**, pp. 319-335.

[10] Y. Wu, Z. J. Mao, S. Fahmy and N. B. Shroff, “Constructing Maximum-Lifetime Data-Gathering Forests in Sensor Networks”, IEEE/ACM Transactions on Networking, vol. 18, no. 5, **(2010)**.

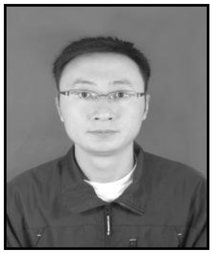
[11] D. J. Luo, X. J. Zhu, X. B. Wu and G. H. Chen, “Maximizing Lifetime for the Shortest Path Aggregation Tree in Wireless Sensor Networks”, In the 30th Conference on Computer Communications, INFOCOM 2011, IEEE Press, Shanghai, China **(2011)**, pp. 1566-1574.

[12] Z. Y. Chen, G. Yang, J. Xu, et al., “A Low-delay Data Aggregation Scheduling Algorithm Based on Leafy Spanning Tree (in Chinese)”, Journal of Nanjing University of Posts and Telecommunications: Natural Science, vol. 32, no. 1, **(2012)**.

[13] J. Könemann, A. Levin and A. Sinha, “Approximating the Degree-Bounded Minimum Diameter Spanning Tree Problem”, Algorithmica, vol. 41, no. 2, **(2005)**.

[14] A. Ghosh, O. D. Incel and V. S. A. Kumar, “Multi-Channel Scheduling and Spanning Trees: Throughput-Delay Trade-off for Fast Data Collection in Sensor Networks”, IEEE/ ACM Transactions on Networking, vol. 19, no. 6, **(2011)**, pp. 1731-1744.

Authors



ZhengYu Chen Dr. Zhengyu chen received Master degree from Nanjing University of Posts and Telecommunications in 2006. He is a Lecture at Jinling Institute of Technology. Currently, He is a Ph.D. candidate in Information Security. His research focuses on information security and privacy preserving. Email: zych@jit.edu.cn.

Geng Yang Dr. Geng Yang received Ph.D. degree in Computer Science from Laval University in 1994 and did a post-doc at Montreal University from 1994 to 1996 in Canada respectively. He is a professor at Nanjing University of Posts & Telecommunications. His research interest includes network security, parallel & distributed computing, mobile computing. Professor Yang is a senior member of the China Computer Federation, a member of the IEEE Computer Society and a Standing Member of Chinese Computer Education Society.

Lei Chen He is currently an associate professor at Nanjing University of Posts and Telecommunications. He received M.S. degree in computer science at Nanjing University of Aeronautics and Astronautics in 2005, and B.S. degree in mathematics from Jiangxi Normal University in 1997. His research interests include services computing, machine learning, neural computing and information security. In these fields, he has authored or coauthored over 20 scientific journal papers.

Jin Wang Dr. Jin Wang received the B.S. and M.S. degree from Nanjing University of Posts and Telecommunications, China in 2002 and 2005, respectively. He received Ph.D. degree from the Computer Engineering Department of Kyung Hee University Korea in 2010. Now, he is a professor in the Computer and Software Institute, Nanjing University of Information Science and technology. His research interests mainly include routing protocol and algorithm design, performance evaluation and optimization for wireless ad hoc and sensor networks. He is a member of the IEEE and ACM.