TDMA scheduling problem avoiding interference in multi-hop wireless sensor networks

*Mihiro SASAKI*, Takehiro FURUTA**, Takamori UKAI*** and Fumio ISHIZAKI****

* Department of Systems and Mathematical Science, Nanzan University
18 Yamazato-cho, Showa-ku, Nagoya, 466-8673, Japan
E-mail: mihiro@nanzan-u.ac.jp

** Teacher Education Center for the Future Generation, Nara University of Education
Takabata-cho, Nara-shi, Nara, 630-8528, Japan

*** Department of Medicine, Tokai University
143 Shimokasuya, Isehara-shi, Kanagawa, 259-1193, Japan

**** Indepa Inc.
2-8-11 Yaesu, Chuo-ku, Tokyo, 104-0028, Japan

Received 7 December 2015

Abstract
In this paper, we consider a multi-hop sensor network, where the network topology is a tree, TDMA (time division multiple access) is employed as medium access control, and all data generated at sensor nodes are delivered to a sink node (the base station) located on the root of the tree through the network. It is reported that if a transmission schedule that avoids interference between sensor nodes completely can be computed, TDMA is preferable to CSMA/CA (carrier sense multiple access with collision avoidance) in performance. In general, the TDMA scheduling problem to find the shortest schedule is formulated as a combinatorial optimization problem, where each combination corresponds to a schedule. However, solving such a combinatorial optimization problem is difficult, especially for large-scale multi-hop sensor networks. The reason of the difficulty is that the number of the combinations increases exponentially with the increase of the number of nodes. In this paper, to formulate the TDMA scheduling problem, we propose a min-max model and a min-sum model. The min-max model yields the shortest schedule, but it is difficult to solve large-scale problems. The min-sum model does not guarantee providing the shortest schedule; however, it may give us good schedules over a short amount of computation time, compared to the min-max model. Numerical examples show that the min-sum model can provide good schedules in a reasonable CPU time, even when the min-max model fails to compute the shortest schedule in a reasonable CPU time.

Key words: Multi-hop wireless sensor network, Scheduling, Integer programming, TDMA, Interference

1. Introduction

With recent rapid advances of wireless technology, wireless sensor networks have been paid much attention due to their rich applications such as environmental, military and health applications. A wireless sensor network consists of wireless sensor nodes, which contain devices for sensing, wireless communication and information processing. Depending on applications, a wide variety of network sizes can be used. In applications such as environmental monitoring, hundreds or even thousands of sensor nodes are deployed in a large monitoring field. For example, Liu et al. (2013) summarize practical sensor network deployment where the number of nodes is from 16 to 200, and conduct a study on a large-scale operating sensor network system called GreenOrbs with 100, 200 and 330 nodes. When sensor nodes are deployed in a field, a network is first organized in an ad-hoc manner. After that, sensor nodes can communicate with each other through the network.

Usually, available resources such as battery power and bandwidth for individual sensor nodes are severely limited. The efficient utilization of the limited resources is thus crucial in wireless sensor networks. This leads to technical
challenges in designing sensor networks. For instance, since a sensor node is driven by a small battery and replacing the battery is not practical, saving the power consumption of sensor nodes is a key to achieve a long lifetime of the sensor network. Hence many researchers have studied the network topology of sensor network to keep the power consumption of sensor nodes low. From this point of view, it is known that a cluster-tree network topology is desirable (see the papers by Culpepper et al. (2004), Heinzelman et al. (2002), Younis and Fahmy (2004) and references therein).

Medium access control, which make it possible for several network nodes to communicate in a multiple access network that incorporates a shared medium, as well as the network topology is a crucial factor to utilize the limited resources of sensor nodes efficiently. Avoiding interference between nodes is one of crucial functions which medium access control in wireless networks should provide. Once interference occurs, packets are not correctly sent and sensor nodes must retransmit the packets. This causes delay and reduces throughput in the sensor network. Furthermore, retransmitting packets additionally consumes the limited battery power of sensor nodes. Kiri et al. (2006) study multi-hop sensor networks, where two or more wireless hops are used to convey information from a source to a destination. They investigate the characteristics of sensor networks by comparing TDMA (time division multiple access) with CSMA/CA (carrier sense multiple access with collision avoidance). They claim that when CSMA/CA is used as medium access control, power consumption increases by 12% and the data collecting time was 3.7 times longer to collect the same amount of packets, compared to TDMA. Thus, TDMA is preferable to CSMA/CA if positional information of all sensor nodes is available and a transmission schedule that avoids interference between sensor nodes completely can be computed. In general, the TDMA scheduling problem is formulated as a combinatorial optimization problem, where each combination corresponds to a schedule (Salcedo-Sanz et al., 2003). However, solving such a combinatorial optimization problem is difficult, especially for large-scale multi-hop sensor networks. The reason of the difficulty is that the number of the combinations increases exponentially with the increase of the number of nodes. Therefore, a technique to find optimal or near-optimal solutions for the TDMA scheduling problem within a practical CPU time should be developed.

In this paper, we consider a multi-hop sensor network, where the network topology is a tree, TDMA is employed as medium access control, and all data generated at sensor nodes are delivered to a sink node through the network. Such a multi-hop sensor network is suitable for applications such as environmental monitoring. For the multi-hop sensor network, we first formulate a TDMA scheduling problem as a min-max model, where the objective is to minimize the last transmission time. Although the min-max model yields the shortest time schedule, it is difficult to calculate the solution of the TDMA scheduling problem because it generally has a large number of tie solutions. To overcome the difficulty, we formulate the problem as a min-sum model, where the objective is to minimize the sum of the time slot numbers when sensor nodes transmit. We then attempt to solve those problems by an optimization software and examine computation times to solve them.

There is a couple of existing studies applying mathematical programming techniques to sensor networks (see the paper by Sorokin et al. (2009) and references therein). However, most of them focus on positioning of sensor nodes, sensing schedule, network connectivity and so on. For packet radio networks with interference between nodes, Salcedo-Sanz et al. (2003) consider the broadcast scheduling problem and develop a mixed neural-genetic algorithm to find a near-optimal solution for the broadcast scheduling problem. To the best of authors’ knowledge, there is no existing work which considers sensor networks with interference between sensor nodes and applies mathematical programming techniques to find near-optimal solutions for the TDMA scheduling problem in the sensor networks.

The remainder of this paper is organized as follows. Section 2 describes the model of the multi-hop sensor network. In Section 3, we present the min-max model and the min-sum model for the formulation of the scheduling problem. In addition, we develop a greedy algorithm to find an upper bound of the shortest time schedule, which enables us to fairly reduce the computational efforts for solving the scheduling problem. In Section 4, we provide some numerical examples of schedules obtained by the min-max model and those obtained by the min-sum model, and we also compare their CPU times. Conclusion is drawn in Section 5.

2. Model Description

In this paper, we consider a centralized multi-hop sensor network where the network topology is a tree (Figure 1). In the centralized model, there is a special facility called base station (BS) located far from or within the monitoring field depending on application. The BS organizes a network and controls all sensors’ activities in the centralized network. Given the tree topology organized by the BS, the BS informing each sensor node of its parent node (the sensor node which it should send a packet to) and child nodes (the sensor nodes which it should receive packets from).
The aim of the sensor network under consideration is to periodically collect all data generated by sensor nodes at a special node called BS. All sensor nodes monitor events and generate data periodically. The generated data is delivered in packet to the BS through the tree network in the following way: A sensor node having child nodes must wait to send a packet including data generated by itself to its parent node until it receives data from all its child nodes. After receiving data from all its child nodes, the sensor node aggregates all data received from the child nodes and data generated by itself in one packet, and it sends the packet to its parent node. This process is repeated, and all data generated at sensor nodes are finally collected at the BS, which is located on the root of the tree. We call the elapsed time to collect all data generated by sensor nodes at the BS as collecting time. Achieving a short collecting time is crucial, especially, for real-time applications.

As for medium access control, we assume that TDMA is employed in the sensor network where time is divided into intervals of equal length, called as time slots, and the transmission time of a packet is one time slot. Since sensor nodes share a wireless channel, a transmission schedule to avoid interference between sensor nodes is required. Once interference occurs, packets are not correctly sent and sensor nodes must retransmit the packets. This causes delay and reduces throughput in the sensor network. Avoiding interference is important in order to save the limited battery power of sensor nodes, too. We assume that packet transmissions always succeed as long as no interference occurs.

We here specify the condition that an interference occurs between sensor nodes in our model. In our model, the tree network organized by the BS can be considered as a subgraph of the unit-graph (Hou et al., 2005). The unit-graph model is widely accepted as a basic graph-theoretical model for wireless sensor networks. In the unit-graph model, all sensor nodes are assumed to use the same and fixed transmission radius \( r \). By establishing arcs between nodes if their distance is less than or equal to \( r \), a unit-graph is composed. Two nodes are called one hop apart with each other if there is an arc between them in the unit-graph. In other words, two nodes can directly communicate with each other, if and only if they are one hop apart. If two nodes are not one hop apart but they have a common node which is one hop apart from the two nodes, the two nodes are called two hop apart with each other. Then, the condition that an interference occurs between sensor nodes is specified as follows: Suppose that node \( i \) transmits a packet to node \( j \) in a time slot. If and only if there exists a node which is one hop apart from node \( j \) and transmits a packet other than node \( i \) in the time slot, an interference against the transmission from node \( i \) to node \( j \) occurs.

Figure 2 illustrates an example that an interference against the transmission from node \( i \) to node \( j \) occurs. Suppose that nodes \( i \) and \( j \) are one hop apart with each other and nodes \( j \) and \( k \) are one hop apart with each other. Since node \( k \), which is one hop apart from node \( j \), is transmitting a packet while node \( j \) is receiving a packet from node \( i \), an interference against the transmission from node \( i \) to node \( j \) occurs.

In the subsequent section, we study formulation to obtain TDMA schedules achieving a short collecting time while no interference occurs. More specifically, for the formulation, we propose two models: min-max model and min-sum model. The min-max model provides the shortest collecting time, but needs a large amount of CPU time. The min-sum model does not guarantee providing the shortest collecting time, but needs less amount of CPU time.
Fig. 2 Interference against the transmission from node $i$ to node $j$ occurs, where $r$ is the transmission radius. Since both pairs of sensor nodes $(i, j)$ and $(j, k)$ are one hop apart with each other, an interference occurs against the transmission from node $i$ to node $j$ if node $k$ is transmitting a packet while node $j$ is receiving a packet from node $i$.

3. Formulation

3.1. Nomenclature

In what follows, we provide integer programming formulations for the TDMA scheduling problem in the sensor network. The objective is to determine a schedule that all the data generated at sensor nodes are collected at the BS as quickly as possible under the constraint that no interference occurs.

To formulate the TDMA scheduling problem, we consider two different models that are called min-max model and min-sum model. The min-max model finds a schedule that provides the shortest time to complete all transmissions. However, we do not expect to solve a large scale problem since the problems with a min-max objective function generally has a large number of tie solutions. Hence, we consider a min-sum model as one of the alternatives. This model does not guarantee producing the shortest time schedule, however, it may give us good schedules over a short amount of computation time.

To describe the two models, we begin with the definition of notations.

$N$: the set of sensor nodes. $|N| = n$.

$T$: the set of time slots. $T = \{1, \ldots, |T|\}$.

$r$: the transmission radius.

$d(i, j)$: distance between sensors $i \in N$ and $j \in N$. $d(i, i) = 0$.

$p(i)$: parent node of sensor $i \in N$.

$S_i$: the set of nodes $j \in N$ where $d(i, j) \leq r$.

$D_i$: the set of descendant nodes of $i \in N$.

$A$: the set of nodes $i \in N$ that have at least one child node. $A = \{i \in N \mid |D_i| \geq 1\}$.

In addition, we introduce the following decision variables.

$x_{it}$: 0-1 variable that takes 1 if sensor $i \in N$ transmits a packet at time slot $t \in T$, 0 otherwise.

3.2. Min-max model

The min-max model can be formulated as follows.

[Min-max model]

\[
\begin{align*}
\min & \quad \max_{i \in N} \sum_{t \in T} t x_{it} \\
\text{s.t.} & \quad x_i + x_j \leq 1, \quad i \in N, j \in S_{p(i)} \setminus \{i\}, t \in T, \\
& \quad x_i \leq \sum_{u=1}^{t-1} x_{ju}, \quad i \in N, j \in D_i, t \in T \setminus \{1\}, \\
& \quad \sum_{t \in T} x_{it} = 1, \quad i \in N, \\
& \quad x_{ii} = 0, \quad i \in A,
\end{align*}
\]
The objective (1) is to minimize the collecting time. Constraints (2) prohibit simultaneous transmission of any two sensors deployed within \( r \) to avoid interference. Constraints (3) ensure that each sensor node having child nodes transmits a packet to its parent node after all its descendant nodes complete transmission. Constraints (4) ensure that each sensor node transmits a packet at a time slot between 1 and \( |T| \). Constraints (5) prohibit sensor nodes having child nodes from transmitting a packet at time slot 1. Constraints (6) ensure that all decision variables take 0 or 1.

The objective function in this model is piece-wise linear. To solve a problem with piece-wise linear functions is generally difficult using optimization software. Then we transform the objective function into linear function by replacing the objective (1) with

\[
\min \delta,
\]

where \( \delta \) is a new decision variable, and adding the following constraints.

\[
\delta \geq \sum_{i \in F} tx_{it}, \quad i \in N.
\]

Thus, the problem becomes a linear integer programming problem, which is tractable with optimization software. Yet the problems is generally hard to solve due to many tie solutions and constraints (8) that indicate \( n \) additional constraints are required to have the objective function linear.

### 3.3. Min-sum model

For the min-sum model as one of alternative scheduling models, we can formulate it as the following linear integer programming problem.

[Min-sum model]

\[
\begin{align*}
\text{min} & \quad \sum_{i \in N} \sum_{t \in T} tx_{it} \\
\text{s.t.} & \quad (2), (3), (4), (5) \text{ and } (6).
\end{align*}
\]

The objective of the min-sum model is to minimize the total sum of the time slot numbers at which each sensor node transmits a packet. Suppose that there are ten sensor nodes. For example, a schedule where five sensor nodes transmit a packet at time slot 1, three sensor nodes transmit a packet at time slot 2, and two sensor nodes transmit a packet at time slot 3, then the objective value of this schedule is 17 calculated by \( 5 \times 1 + 3 \times 2 + 2 \times 3 \). In this case, the collecting time is 3. With this objective, each sensor node prefers to transmit a packet as soon as possible after it receives all packets from the child nodes in order to minimize the objective value. From this observation, we expect that the min-sum model produces a schedule that achieves nearly shortest collecting time. In contrast, with the min-max objective, each sensor node does not necessarily transmit a packet as soon as possible unless it is critical to minimize the objective value, i.e., the collecting time.

### 3.4. Greedy algorithm to find an appropriate value of \(|T|\)

The number of time slots \(|T|\) affects the computational effort since the number of variables and constraints vary depending on \(|T|\). In each model, setting the value of \(|T|\) at \( n \ (= |N|) \) is sufficient for the worst case corresponding to that each sensor node transmits a packet one by one in each time slot. However, depending on a given tree and transmission radius, this may be overestimated. Hence we develop a greedy algorithm in order to find an upper bound of the shortest time schedule, which can be an appropriate value of \(|T|\).

The basic idea behind the greedy algorithm is simply finding a set of sensor nodes \( F \) at each time slot in which all sensor nodes are able to transmit data simultaneously without causing interference. In the following algorithm description, \( \bar{N} \) is defined as sensor nodes that have not transmitted yet at the beginning of step 1 in time slot \( \tau \). By iterating step 1, \( \bar{N} \) is divided into two sets \( F \) and \( W \), where the sensor nodes in \( W \) are not transmitting at time slot \( \tau \). Note that \( W = \bar{N} \setminus F \).

The greedy algorithm is described as follows.

[Greedy Algorithm]

**Input:** the set of sensor nodes \( N \), and \( p(i), S_i \), and \( D_i \) for all \( i \in N \).

**Output:** an upper bound \( \tau^* \) and a feasible solution \( x_{it}^* \) for all \( i \in N \) and \( t = 1, \ldots, \tau^* \).

[DOI: 10.1299/jamdsm.2016jamdsm0047] © 2016 The Japan Society of Mechanical Engineers
Step 0: Set $F := \emptyset$, $W := \emptyset$, $\bar{N} := N$, and $\tau = 1$.  
Step 1: If $\bar{N} := \emptyset$ then go to Step 2, else select $j \in \bar{N}$ and set $\bar{N} := \bar{N} \setminus \{j\}$.  
If the following three conditions hold, then set $F := F \cup \{j\}$ and $x^*_t = 1$, otherwise set $W := W \cup \{j\}$ and $x^*_t = 0$. Return to Step 1.  
Step 2: If $W = \emptyset$ then output an upper bound $T^*$, a feasible solution $x^*_t$ for all $i \in N$ and $t = 1, \ldots, T^*$. Terminate. Else, set $\tau := \tau + 1$, $\bar{N} := W$, $F := \emptyset$, $W := \emptyset$ and return to Step 1.

4. Numerical Examples

In this section, we show some optimal solutions of the min-max and min-sum model using test data where the sensor nodes are deployed in a grid-like pattern in a square of 100 by 100 meters. We generate four different unit-graphs with different number of sensor nodes and different transmission radius (Figures 3 and 4). Figure 3 shows a unit-graph where 49 (7 × 7) sensor nodes are deployed in the square, and Figure 4 shows a unit-graph where 100 (10 × 10) sensor nodes are deployed in the square. The sensor networks composed of 49 or 100 sensor nodes are considered as mid-scale sensor networks.

In each data set, we generate two different unit-graphs using different values of transmission radius. In the first unit-graph, each sensor node can communicate with sensor nodes in four directions (Figures 3(a) and 4(a)). In other words, for a sensor except on the edge, there are four neighbors within its transmission radius. We call these two unit-graphs as 49-4D and 100-4D, respectively. Similarly, in the second unit-graph, each sensor node can communicate with sensor nodes in eight directions (Figures 3(b) and 4(b)). We call these two unit-graphs as 49-8D and 100-8D, respectively.

Figures 5 and 6 show examples of trees defined on the unit-graphs with the BS indicated by black square. These are shortest path tree (SPT) where the distance between the BS to all other nodes through the unit-graph is minimum. Note that the coordinates of sensor nodes are shifted from the right position of the grid using random small values so as to have an unique SPT on the unit-graphs.

We used an optimization software IBM ILOG CPLEX 12.1 to obtain exact optimal solutions of min-max models and min-sum models. All experiments were performed on a PC with Intel Core2 Duo CPU (2.53 GHz) and 2014 MB RAM. Figures 7, 8, 9, and 10 show optimal schedules using data set 49-4D, 49-8D, 100-4D, 100-8D, respectively. The encircled numbers show the time slot each node transmits a packet. Note that we do not obtain a min-max solution with the unit-graph 100-8D due to memory shortage.

Table 1 summarizes the comparisons of the results of the min-max model and the min-sum model. As mentioned in Section 3.4, the greedy algorithm provides us an upper bound of the shortest collecting time. Hence we introduce the upper bounds into the min-max and min-sum models using them as a value of $|T|$. The second column in Table 1 labeled “$|T|$ (Greedy)” shows the collecting times obtained from the greedy algorithm, which are used as the value of $|T|$ in the two models.

Although we introduce an efficient upper bound, we still do not obtain a min-max solution with the unit-graph 100-8D. In each of the other cases, i.e., 49-4D, 49-8D and 100-4D, there is no difference between the collecting times.
Table 1  Computational results

| Unit-graph | [T] (Greedy) | Collecting time | CPU time (sec.) |
|------------|--------------|-----------------|-----------------|
|            | Min-max      | Min-sum         | Min-max         | Min-sum         |
| 49-4D      | 10           | 10              | 10              | 0.19            | 0.17            |
| 49-8D      | 15           | 13              | 13              | 930.34          | 7.17            |
| 100-4D     | 13           | 13              | 13              | 0.31            | 0.08            |
| 100-8D     | 18           | –               | 16              | –               | 705.30          |

Fig. 5  The SPTs of 49 sensor nodes (Black square is the BS.)  
Fig. 6  The SPTs of 100 sensor nodes (Black square is the BS.)

Fig. 7  Transmission schedules of each node with unit-graph 49-4D.  
Fig. 8  Transmission schedules of each node with unit-graph 49-8D.

Fig. 9  Transmission schedules of each node with unit-graph 100-4D.  
Fig. 10  Transmission schedules of each node with unit-graph 100-8D.
obtained by the min-max model and the min-sum model. This means that we obtained optimal schedules for the three cases using the min-sum model. In addition, the greedy algorithm also finds optimal schedules with the unit-graphs 49-4D and 100-4D.

For the CPU time, we solved min-sum models much faster than min-max models. Min-sum models may be useful to solve larger problems even though they do not guarantee providing a schedule completed in exact shortest time.

We need large amount of computation time for solving problems defined on eight-direction unit-graphs. The scheduling problem with eight-direction unit-graph is more complicated than that with four-direction unit-graph because we need to consider interference from the double number of sensor nodes in each transmission.

5. Conclusion

In this paper, we considered a multi-hop sensor network, where the network topology is a tree, TDMA is employed as medium access control, and all data generated at sensor nodes are delivered to the BS node located on the root of the tree through the network. To solve the scheduling problem for TDMA, we presented new formulation where the problem is formulated as integer programming problems. In particular we proposed the min-max model and the min-sum model. While the min-max model finds the shortest schedule satisfying the constraints, the numerical examples show that it is hard to get the solution within a practical CPU time. The reason is that the problem with the min-max objective function generally has a large number of tie solutions. Hence, we proposed a min-sum model as one of the alternatives. While this model does not guarantee producing the shortest time schedule, it yields good schedules over a short amount of CPU time. In future research, we intend to develop heuristics to solve larger problems in practical CPU time.

Acknowledgment

The authors appreciate Dr. Chaya Losada for her helpful comments during the early stage of the work, and the anonymous referees for their valuable comments that have helped improve the paper. The work was partially supported by Nanzan University Pache Research Subsidy I-A-2 for the 2016 academic year.

References

Culpepper B.J., Dung L. and Moh M., Design and analysis of hybrid indirect transmissions (HIT) for data gathering in wireless micro sensor networks, ACM Mobile Computing and Communications Review, Vol. 3, No. 1 (2004), pp. 61-83.
Heinzelman W.B., Chandrakasan A.P. and Balakrishnan H., An application-specific protocol architecture for wireless microsensor networks, IEEE Transactions on Wireless Communications, Vol. 1, Issue 4 (2002), pp. 660-670.
Hou J.C., Li N. and Stojmenović I., Topology construction and maintenance in wireless sensor networks, In I. Stojmenović, editor, Handbook of Sensor Networks: Algorithms and Architectures (2005), John Wiley & Sons, Inc., pp. 311-341.
Kiri Y., Sugano M. and Murata M., Performance evaluation of intercluster multi-hop communication on large-scale sensor networks, Proceedings of International Conference on Computer and Information Technology (2006), p. 215.
Liu Y., He Y.H., Li M., Wang J., Liu K. and Li X., Does wireless sensor network scale? a measurement study on GreenOrbs, IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 10 (2013), pp. 1983-1993.
Salcedo-Sanz S., Bousono-Calzon C. and Figueiras-Vidal A.R., A mixed neural-genetic algorithm for the broadcast scheduling problem, IEEE Transactions on Wireless Communications, Vol. 2, Issue 2 (2003), pp. 277-283.
Sorokin A., Boyko N., Boginski V., Uryasev S. and Pardalos P.M., Mathematical programming techniques for sensor networks, Algorithms, Vol. 2, Issue 1 (2009), pp. 565-581.
Younis O. and Fahmy S., Distributed clustering in ad-hoc sensor networks: A hybrid energy-efficient approach, Proceedings of IEEE INFOCOM (2004), pp. 629-640.