PAPER *Special Section on Ubiquitous Sensor Networks*

Data Transmission Scheduling Based on RTS/CTS Exchange for Periodic Data Gathering Sensor Networks

Takashi MATSUDA[†a)], **Masumi ICHIEN**[†], **Hiroshi KAWAGUCHI**[††], ***Nonmembers***, **Chikara OHTA**[††], ***and*** **Masahiko YOSHIMOTO**[††], ***Members***

SUMMARY One challenging issue of sensor networks is extension of overall network system lifetimes. In periodic data gathering applications, the typical sensor node spends more time in the idle state than active state. Consequently, it is important to decrease power consumption during idle time. In this study, we propose a scheduling scheme based on the history of RTS/CTS exchange during the setup phase. Scheduling the transmission during transfer phase enables each node to turn off its RF circuit during idle time. By tracing ongoing RTS/CTS exchange during the steady phase, each node knows the progress of the data transfer process. Thereby, it can wait to receive packets for data aggregation. Simulation results show a 160–260% longer system lifetime with the proposed scheduling scheme compared to the existing approaches
key words: *sensor network, data gathering, scheduling, cross-layer*

1. Introduction

Recently, advances in micro-sensors, low-power electronics, and low-power radio frequency design have enabled the development of small and low-power sensor nodes. Wireless sensor networks comprising such sensor nodes have attracted attention for extracting information from a wide range of sources. Generally, the sensor node works with a battery. Virtually, all sensor nodes use batteries so decreasing power consumption is necessary to reduce the battery exchange frequency. Therefore, extension of the system lifetime is the hurdle to achieve wireless sensor networks.

In order to lengthen the system lifetime, development of low power wireless communication for wireless sensor networks is one of the most important issues because it is largest consumer of power. In particular, the following factors should be considered:

- Idle listening: The RF circuit operates while the sensor node is not communicating. The RF circuit consumes as much power when the sensor node does not receive data as when it does.
- Retransmission by collision: The sensor node should retransmit data and will consume power when the data transmission fails because of collision.

In this study, we target the periodic data gathering applications, which periodically observe such as temperature and humidity. In this kind of applications, the base station collects sensed information from the sensor nodes. In order to extend the system lifetime for this kind of applications, we approach cross-layer design of the application, network, and MAC (Media Access Control) layer for the above issues.

As described in Sect. 2, some protocols have been proposed for the periodic data gathering. Among them, Power Scheduling (PS) is promissing in the sense that it requres neither complicated MAC nor global knoweledge of the networks. PS needs only an MAC with the RTS/CTS (Request To Send/Clear To Send) exchange function such as MACA (Multiple Access Collision Avoidance) [8] and PAMAS (Power function Aware Multi-Access protocol with Signaling) [6].

In PS, the network tree is composed for data collection (Fig. 1), and it is recycled two or more times. Each node records the timing of the successful data transmission in the first data gathering after tree construction, and then it repeatedly transmits data packets at the same timing in the following data gathering. In this sense, PS is a time-based scheduling scheme. From now on, the duration of the first data gathering and that of the following data gatherings are called "setup phase" and "steady phase," respectively. In the steady phase, the series of the successful data transmissions in the setup phase is repeated so that packet collision is avoided in the steady phase. Therefore, RTS/CTS exchange is unnecessary in the steady phase, which saves power consumption. Further, each node has only to wakeup for scheduled data transmissions and receptions so that the

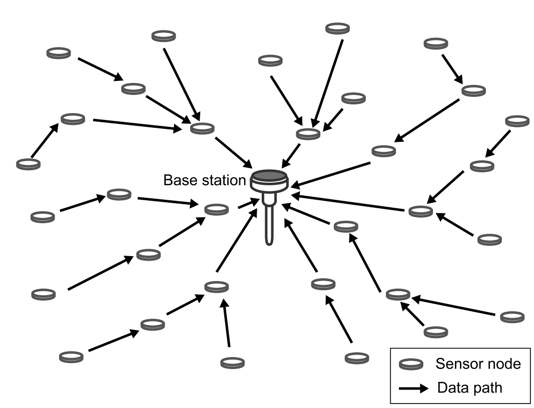


Fig. 1 Sensor network tree for data gathering application.

Manuscript received March 30, 2007.
Manuscript revised June 30, 2007.
[†]The authors are with the Graduate School of Science and Technology, Kobe University, Kobe-shi, 657-8501 Japan.
[††]The authors are with the Faculty of Engineering, Kobe University, Kobe-shi, 657-8501 Japan.
a) E-mail: matsuda@cs28.cs.kobe-u.ac.jp
DOI: 10.1093/ietcom/e90–b.12.3410

idle listening can be suppressed.

Time-based scheduling in PS, however, leads to the following three drawbacks: First, time synchronization among nodes should be kept accurate. To do so, a synchronization protocol such as .... is necessary to be implemented. Second, data packets can not be retransmitted in the steady phase so as not to disarrange shceduling timing even if they are received incorrectly. In the case where MAC layer supports retransmissions, an enough margin should be reserved in the schedule for serveral retries. Third, for similar reasons, the size of sensed data in the steady phase should not exceed that in the setup phase so as not to upset schedule timing. In the case of variable-legnth data, a sufficient margin should be afforded in the schedule.

In addition to the above drawbacks, data aggregation [10] is not considered enough. That is, PS does not provide any function to facilitate data aggregation. Therefore, each node may not completely aggregate incoming packets although it may happen to aggregate received packets before relaying them towards the base station. If data aggregation is used, the length of outgoing packets after aggregation should be invariant packet by packet. Therefore, not every data aggregation scheme is applicable.

To cope with these problems, an event-based scheduling scheme, named "Multi-layer Scheduling (MS)," is proposed as an improved PS in this paper. In MS, the RTS/CTS exchange is used even in the steady phase unlike in PS. This enables each node to trace the progress of the data transfer around itself. As a result, each node can decide when it should send data packets. Because the interval between successive events is variable, nodes can postpone to transmit data packets so that they can fully exploit data aggregation to the greatest possible extent as described in detail in Sect. 3. Apparently, the usage of the RTS/CTS exchange brings the increased power, but it can be mitigated by means of PAMAS (Power function Aware Multi-Access protocol with Signaling) [6] and the rewards of data aggregation outweighed the overhead of the RTS/CTS exchange in the steady state as shown in 4.

2. Related Works

Some protocols for periodic data-gathering applications have been studied: Low-Energy Adaptive Clustering Hierarchy (LEACH) [1], [11], Power-Efficient GAthering in Sensor Information Systems (PEAGASIS) [3], [9], Power Scheduling (PS) [2], etc.

In LEACH, a small number of clusters are formed in a self-organized manner. A cluster head collects and fuses sensed data from the cluster member nodes inside its cluster, and transmits the result to the base station. LEACH saves power by using transmission power control and data fusion. However, in order to form clusters and avoid radio interference among them, LEACH requires an MAC protocol composed of CSMA, CDMA and TDMA, which makes the RF design of sensor nodes complicated.

In PEGASIS, in order to exploit the effect of transmission power control and data fusion, the sensor nodes form a chain so that each node communicates with a closest neighbor. If each node fuses received data perfectly, in other word, perfect aggregation is performed [4], the total transmission power over all the nodes is minimized because each node transmits data to a closest neighbor. However, perfect aggregation limits available applications. Further, PEGASIS requires that all nodes should have global knowledge of the networks.

Power Scheduling (PS) is a scheduling scheme for periodic data gathering [2]. PS only needs an MAC with an RTS/CTS exchange function. Unlike the above, however, PS requires neither complicated MAC nor global knowledge of the networks. Therefore, we focus on PS in the paper.

The PS scheme lowers power consumption by scheduling the send and receive times and turning off the RF circuit when not either sending or receiving. The PS scheme has two phases: the setup phase and the steady phase. In setup phase, sensor nodes transfer their sensed data to the base station using CSMA with RTS/CTS exchange. At this time, each sensor node records successful data transmission and reception with their occurrence time in its table. In the steady phase, the sensor nodes recreate such successful events on time according to their table. And for the rest of time, they go to sleep. Moreover, because only successful events are repeated, packet collisions do not occur at all. So, power scheduling is expected to extend the system lifetime.

First, sensor nodes communicate a data packet using CSMA with RTS/CTS exchange in the setup phase. In this regard, the sensor node records the communication beginning time in the table when the data packet is sent correctly. Here, the communication beginning time that is recorded in the table is the transmission beginning time of the RTS packet. Figure 2 shows an example of a schedule in the case of sensor nodes sending their own sensing data to a base station using a multi-hop. In this table, the character on the left of each element shows the event type: the sender transmits data (STD) and the receiver receives data (RRD). The right figure shows the communication beginning time. The sensor

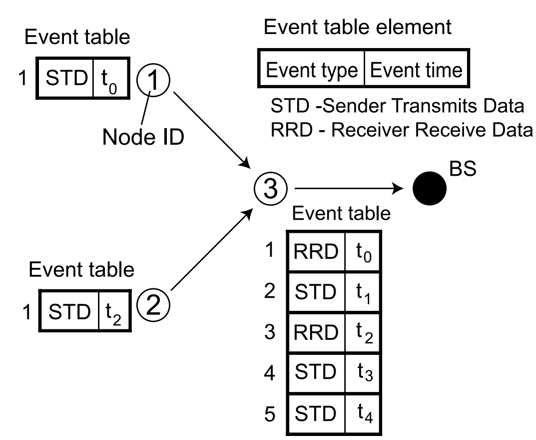


Fig. 2 A 3-node sensor network with a base station and an exemplary schedule by power scheduling.

node sends data after the t_0 seconds from the data collection beginning time if the element in the table is STD-t_0. The element's addition to the table is according to the following.

- When the node transmits: The sensor node records the transmission event (STD) and the transmission beginning time.
- When the node receives: The sensor node records the reception event (RRD) and the reception beginning time.

Figure 2 depicts the flow of the schedule-making. First, node 1 transmits a data packet to node 3 so that node 1 takes sensing data to the base station. At that time, node 1 records the information of sending data (STD) and the transmission beginning time (t_0) in the table (node 1 - table 1). Then, node 3 records the receiving data (RRD) information and the reception beginning time (t_0) in the table (node 3 - table 1). Actually, a propagation delay pertains between the sending and receiving time. However, for simplicity, we show it as the same time. Next, node 3 relays the node 1 data to the base station. Node 3 records the information of sending data (STD) and the transmission beginning time (t_1) in the table (node 3 - table 2). These nodes add events to the table as well as ahead when node 2 sends its own sensing data to the base station.

Next, in the steady phase, no sensor nodes use RTS/CTS exchange because the transmission time of each node is decided. To the greatest degree possible, sensor nodes turn off the RF circuit power. This sleep process turns off the RF circuit of the sensor node. The sensor node turns on the RF circuit immediately before the transmission beginning time of the table that was made in the setup phase; it sends data immediately. After the sensor node sends data, it sleeps until the time of the next gathering period. The sensor node turns on the RF circuit immediately before the time of the reception time that is recorded in the table. The sensor node then sleeps until the time of the following event when the sensor node finishes receiving data. In that manner, the PS scheme extends the system lifetime by obviating idle listening.

However, because of time-synchronization gaps among sensor nodes, it might not be possible for them to communicate with each other. For example, if the inner clock of node 1 shifts in relation to those of other nodes, node 1's transmissions would interfere with those of other nodes. In the PS scheme, sensor nodes cannot transmit again because sensor nodes do not know the schedules of other nodes. The PS method does not positively aggregate data. It is necessary to prepare a special node that can serve the same role as the base station if data aggregation is needed.

The PS scheme has three drawbacks. One is that the PS scheme cannot retransmit data because retransmission would jumble the schedule and thereby create a data transmission error. Another is that synchronization must be maintained among nodes because of its use of time-based scheduling. Finally, the PS scheme might not make the best use of its possible aggregation. If a route is consistently available to send data to the base station, each node should retain a schedule that is decided once, and then perform transmissions separately. If in-network aggregation is used, the same route set procedure can be used while replacing the base station with node that perform the aggregation. However, when the sensor node does taking the place of the base station, the battery of its sensor node is dead quickly.

3. Multi-Layer Scheduling

We propose Multi-layer Scheduling (MS), which schedules based on RTS/CTS exchange [13]. The MS scheme has two phases that resemble phases of the PS scheme: the setup phase and steady phase. The outline of these phases is as follows.

Setup phase: The sensor nodes communicate a data packet using CSMA with RTS/CTS exchange and record RTS/CTS receive event in addition to STD and RRD. Moreover, the sensor nodes modify its event table to aggregate data packets in the steady phase.
Steady phase: Each sensor node decides when it sends data based on not time but progress of events recoded in its event table that was produced in the setup phase. When sensor nodes send data, the sensor nodes also use RTS/CTS exchange.

Two major differences exist with the PS method. One is the ability to retransmit data because of event-type scheduling and RTS/CTS exchange at the steady phase. Another is the ability to aggregate data of all sensor nodes.

Both the PS scheme and MS scheme are independent of the route-decision method. However, in this work, sensor nodes collect data to the base station in accordance with Tiny Diffusion, which is a simplified Directed Diffusion method [5], [7]. In Tiny Diffusion, the base station broadcasts an interest packet to the entire network. Each sensor node that is targeted for the interest packet sends sensed data to the base station. In our target application, we assume that all sensor nodes are targeted for the interest packet and send data to base station. The MS method procedure is shown as follows.

3.1 Setup Phase

First, the base station broadcasts an interest packet to the entire network, which has an interval of data collection and a data collection frequency in steady phase. Moreover, the interest packet has the number of minimum hops from the base station. The base station sets the initial value to zero. This means that the base station is zero hop away from itself. The sensor node receiving the first interest packet updates the number of minimum hops by adding one and forwards it to neighbor nodes. The sensor node compares the number of minimum hops in the interest packet with that of itself when the sensor node receives two interest packets or more. The sensor node updates the number of minimum hops and forwards the interest packet again if the number of minimum hops in the interest packet plus one smaller than that of the sensor node. If that is not so, then the sensor node does not forward the interest packet, thereby limiting the volume of interest traffic.

Next, each source node sends sensed data to the base station or to a relay node that is nearer the base station. The relay node is that which can send the interest packet to the source node earliest with minimum hops. The node relays it to its relay node, as described above, when the node receives a packet from the other node. In the first collection, sensor nodes transmit sensing data simultaneously because it is not understood whether there are relay data or not. From the second time of collection, sensor nodes can aggregate data because they know when the last of the relay data arrives. This method is described in paragraph 3.2.

In the MS scheme, sensor nodes determine the timing to send a packet based on RTS/CTS packets used in this MAC protocol. Sensor nodes receive RTS/CTS packets with the recorded packet type, with the transmission/reception beginning times of these packets, and the sender and receiver addresses in the table. Figure 3 shows an example of table production of the node for transmission from node 2 to 3. In this case, we assume nodes 1 and 3 can receive data form node 2, and nodes 2 and 4 can receive data form 3. The character at the left of each element in this table shows the event type, which is one of the following: other receives RTS (ORR), other receives CTS (ORC), sender transmits data (STD), and receiver receives data (RRD). The second and third figures from the left of each element respectively show sender-node and receiver-node addresses. The rightmost figure of each element shows the transmission or reception beginning time of the packet. The condition of adding the event to the table is as follows.

- When the node transmits data: The sensor node records the transmission data event (STD) and the beginning time of transmission (node 2).
- When the node receives data: The sensor node records the reception data event (RRD) and the beginning time of reception (node 3).
- When the node receives RTS: The sensor node records the reception RTS event (ORR) and the beginning time of reception (node 1).
- When the node receives CTS: The sensor node records the reception CTS event (ORC) and the beginning time of reception (node 4).

If a sensor node receives the above events again from the same sender, the sensor node deletes the old event and adds a new event to the end of the table. This is done to send data to the next node that are aggregated with other data from further nodes from base station in the steady phase. Moreover, the sensor node turns the radio off after sending data; it subsequently deletes events in the table.

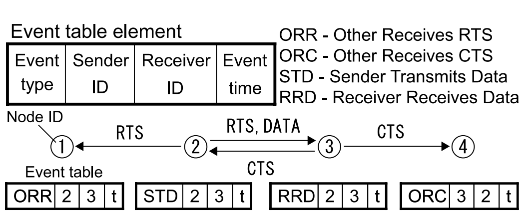


Fig. 3 Making list at each node when node B sends to node C.

Figure 4 portrays an example of table-making using the MS method. First, node 1 sends its sensed data to node 3 for subsequent relay to the base station. Node 1 exchanges a RTS/CTS packet with node 3. Node 2 receives this CTS packet from Node 3, and records this information of the CTS packet in the table (node 2 - table 1). Node 1 and node 3 also respectively record transmit and receive information of the table (nodes 1 and 3 - table 1). Next, node 3 forwards node 1's data to the base station. In this case, node 3 exchanges an RTS/CTS packet with the base station similarly, and node 1 and node 2 receive the RTS packet from node 3. Then, node 1 and node 2 record CTS information in each table (node 1 and 2 - table 2). Node 3 records information of sending data in the table (node 3 - table 2). Similarly, node 2 sends its sensed data to node 3 to be relayed to the base station. In this case, node 2 exchanges an RTS/CTS packet with node 3 as well as ahead. Node 1 receives node 3's CTS packet and records this information in the table (node 1 - table 3). Nodes 2 and 3 record transmit and receive information in the table (nodes 2 and 3 - table 3). Next, node 3 forwards node 2's data to the base station. Node 1 and node 2 receive the RTS packet from node 3. Nodes 1 and 2 have received the RTS packet from node 3 before, so they delete old information (node 1 and 2 - table 2) and add new information (nodes 1 and 2 - table 4) in the table. Node 3 also deletes old information (node 3 - table 2) and adds transmit information in the table (node 3 - table 4). Finally, node 3 transmits its own sensed data to the base station. Nodes 1 and 2 delete old information (nodes 1 and 2 - table 4) and add new information (node 1 and 2 - table 5) to the table again. Node 3 deletes old information (node 3 - table 4) and adds transmit information to the table (node 3 - table 5). Each sensor node maintains this table until the next setup phase.

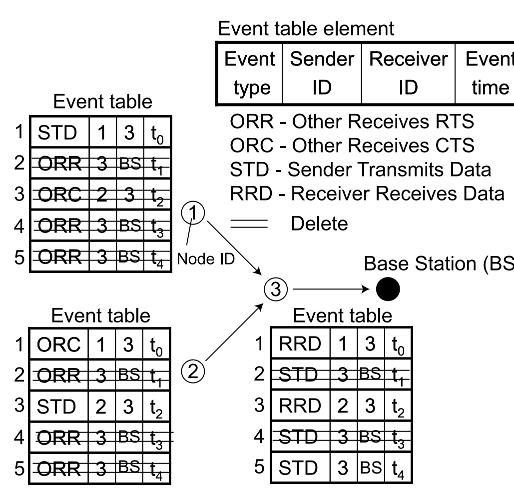


Fig. 4 Example of a 3-node sensor network with a base station and a possible schedule using multi-layer scheduling.

3.2 Steady Phase

In the steady phase of the MS scheme, each sensor node sends data based on its table that was produced in the setup phase. First, sensor nodes with elements of data transmission at the top of the table send data. In the MS scheme, sensor nodes exchange RTS/CTS packets differently from PS scheme, so neighbor nodes receive RTS/CTS packets and check elements the table. Sensor nodes that have transmission as the unchecked top element in the table prepare to send data shortly after sleep by RTS/CTS. Nodes that finish transmission sleep until the next data collection.

Figure 4 shows an illustrative example of the MS method flow. First, we discribed the procedure for establishing of the link. The base sation transmits interest packet to node 3. Node 3 forwards interest packet to node 1 nad 2. Then, sensor nodes transmit sensed data following the reverse of interest packet flow. Next, we discribed the procedure for making event table. Node 1, with the data transmission element at the top of the table among three nodes, sends data to node 3. In the MS method, node 1 exchanges an RTS/CTS packet with node 3, and node 2 receives the CTS packet. Then nodes 1, 2 and 3 check the first element in the table. After node 1 finishes sending data, node 1 goes to sleep until the next data collection. Next, node 2 with unchecked top element of transmission in its table sends its sensed data to node 3. Then nodes 2 and 3 check the second element in their respective tables. After that, node 2 goes to sleep. Finally, node 3 sends aggregated data to the base station and sleeps until the next data collection. In this manner, sensor nodes aggregate data and send data smoothly with avoiding packet collisions.

According to the above procedure, some nodes may keep listening for a long time to receive the relay data. This problem possibly happens in the networks with large diameter where the number of hops from the base station is large. To overcome this problem, if a sensor node has enough time until the expected time to receive relay data in the steady phase, a sensor node lets itself sleep for as long as possible. To do so, the threshold T_{th} is introduced in this paper. Here, let t_{r} denote the expected time of relay data reception listed in the event table, and let t_{c} denote the current time. If $T_{\mathrm{th}} < t_{\mathrm{r}} - t_{\mathrm{c}}$, the sensor node goes to sleep until $t_{\mathrm{r}} - T_{\mathrm{th}}$.

Sensor nodes cannot check the table and send data if sensor nodes cannot receive RTS/CTS packets because of interference or collision. Then, if the sensor node cannot send data at the previous transmit time and receive all data that should be relayed, the sensor node aggregates these data and sends them to next hop node immediately. If the sensor node was unable to send data at the previous transmit time and did not receive all relay data, the sensor node sends aggregated data to the next hop node instantly. This node waits some time after it sends data. If the node still does not receive new relay data, it then goes to sleep.

In the MS scheme, sensor nodes exchange RTS/CTS packets in steady phase unlike the PS scheme, so sensor nodes can retransmit data to avoid collision. On the other hand, in the PS scheme, the sensor node cannot retransmit data for its time schedule and collision.

4. Performance Evaluation

We used QualNet simulator to evaluate our scheduling.

4.1 Simulation Conditions

We compare three schemes (MS Scheme, PS scheme, and no scheduling) in three different cases:

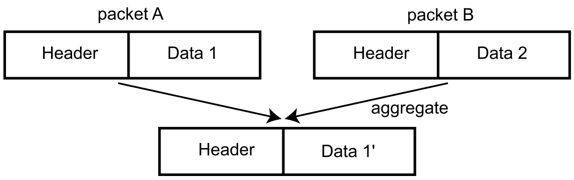
(a) 100 sensor nodes in a 100 m × 100 m rectangular area.
(b) 200 sensor nodes in a 100 m × 100 m rectangular area.
(c) 225 sensor nodes in a 150 m × 150 m rectangular area.

Here, the density of the sensor nodes in case (b) is twice as high as that in case (a), and the maximum number of hops from the base station in case (c) is 1.5 times as large as that in case (a). Sensor nodes were deployed in a sensing area uniformly. For each parameter setting, 20 trials with different random seeds were executed and the average value of them are plotted in the following graphs. The transmission range is assumed to be circular with a 20 m radius. We assume that the transmission power is 800 μW, the reception power is 800 μW, and the sleep power is 0.5 μW. We assume that the battery capacity of each sensor node is 500 mJ. We assume that the header of each packet is 32 bytes. The packet payload is 16 bytes. The RTS/CTS packet size is 24 bytes. The ACK packet size is 20 bytes. The bit rate is 10 kbps. The sample period of the Low-Power Listening (LPL) is 50 ms. The data gathering period is 900 s. We assume that header aggregation eliminates headers of two or more packets for the same destination. The base station broadcasts an interest packet every 10 rounds and reconfigures the routing tree. Signal to noise ratio (SNR) threshold is 10 dB. If SNR threshold exceeds 10 dB, In is considered that the channel interference occured. in this case, a sender node cannot receive ACK packet from a receiver node and sender node retransmits data.

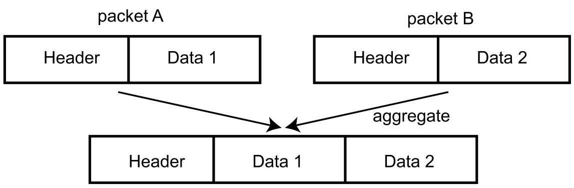
Next, we explain the aggregation method used for this simulation. Two methods of data aggregation exist: perfect aggregation and linear aggregation. The features of each aggregation method are as follows (Fig. 5) [4].

Perfect aggregation: For applications that require such as maximum, minimum, and mean of sensor measurements over all the sensor nodes, sensor nodes receive all packets, operate, and forward one packet. Although this method is used by many proposed protocols, it is not practicable because applications are limited.
Linear aggregation: The sensor node aggregates data to the same target node in one packet and forwards it. The sensor node then deletes the header of each datum and brings them together, thereby reducing traffic. It is possible to apply this to any application.

The MS scheme does not depend on the aggregation



(a) Perfect aggregation.



(b) Linear aggregation.

Fig. 5 Data aggregation type: Perfect aggregation and linear aggregation.

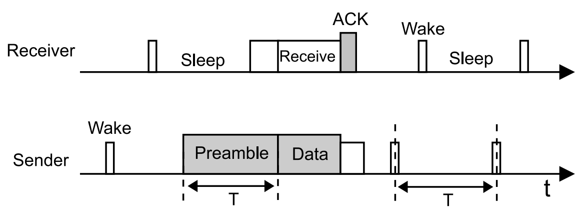


Fig. 6 Timing chart of low-power listening.

method. We validate it using linear aggregation with wide applications in this simulation. The MS scheme, which positively aggregates data, is more advantageous than the PS scheme when it is possible to aggregate data completely.

The MS scheme uses LPL as its medium access control for data communication to reduce power consumption. Figure 6 shows the LPL operation. In LPL, the radio periodically samples the channel. If there is nothing to receive, the radio powers off, otherwise it wakes up from LPL mode to receive the incoming packet. For that reason, the radio on-time is short. Therefore, the power consumption of the LPL method is very low. If the sensor node needs to send data, the sensor node sends a preamble that is as long as the radio channel sampling interval (T). Then the neighbor nodes contain destination node wake up radio to prepare to receive incoming packet. However, because the sensor node sends data immediately after the preamble, it may cause a hidden terminal problem and overhearing. Consequently, we use LPL with the RTS/CTS exchange function to obviate that problem. Figure 7 shows the operation of LPL with RTS/CTS exchange.

The sensor nodes that receive the RTS/CTS packet turn the radio off for the duration described in RTS/CTS packet, as in Power Aware Multi-Access protocol with Scheduling (PAMAS) [6]. In MS scheme, RTS/CTS exchange in steady phase possibly increases the amount of control packets so that the system life time may be shorten. To mitigate this problem, PAMAS is effective. In PAMAS, neighbor nodes that received RTS/CTS packets sleep; the time is included in RTS/CTS packet. Consequently, the power consumption is

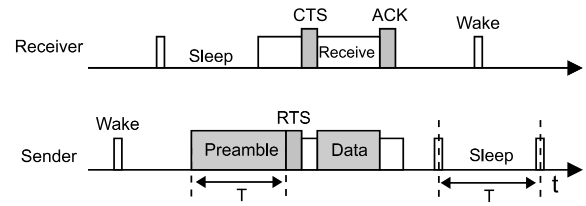


Fig. 7 Timing chart of low-power listening with RTS/CTS exchange.

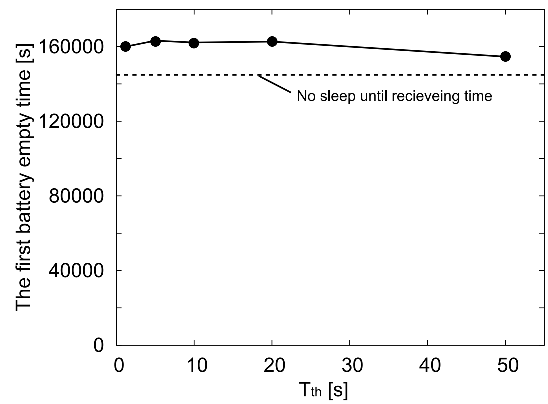


Fig. 8 The lifetime of MS scheme changing the value of T_{th} in case (a) 100 sensor nodes in a 100 m × 100 m rectangular area.

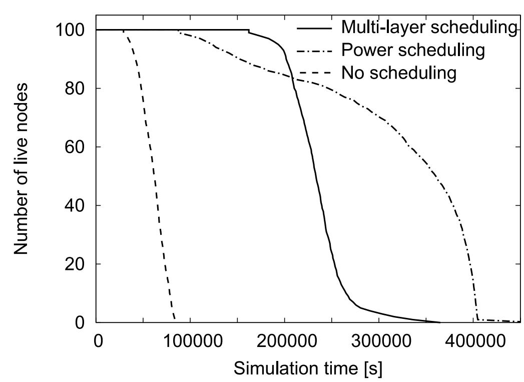


Fig. 9 The lifetime in 100 sensor nodes in a 100 m × 100 m rectangular area (a).

reduced because sensor nodes can avoid listening activities.

4.2 Simulation Result

First, we simulated MS scheme in case (a) while changing the value of T_{th}. In Fig. 8, straight line shows the lifetime when sensor nodes do not sleep until receiveing time. When T_{th} is small, the lifetime is not so improved. Beacause of the disorder of schedule, retransmission has happened frequently. To the contrary, When T_{th} is large, the time that sensor node sleeps decrease. To reflect this simulation, we set T_{th} ten seconds in the following simulation.

Figure 9 shows the number of nodes that are alive by simulation time for the MS scheme, the PS scheme, and a non-scheduling scheme in case (a). As this figure shows, the MS scheme has the longest time of all nodes live among the

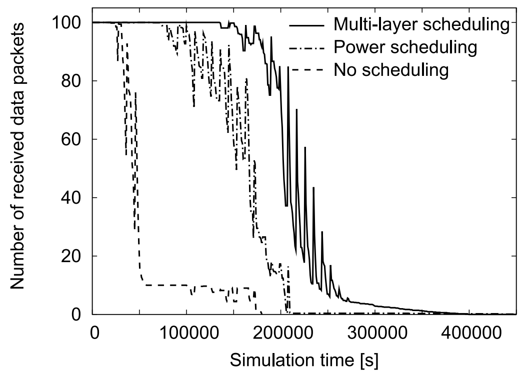


Fig. 10 Number of receive data packets in 100 sensor nodes in a $100\,\mathrm{m} \times 100\,\mathrm{m}$ rectangular area (a).

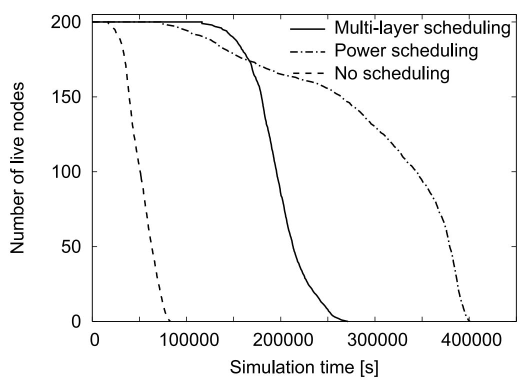


Fig. 11 The lifetime in 200 sensor nodes in a $100\,\mathrm{m} \times 100\,\mathrm{m}$ rectangular area (b).

three schemes. The PS scheme reverts to the MS scheme at 210,000 s. However, sensor nodes near the base station typically run down their batteries, so the base station becomes unable to receive data from distant nodes (Fig. 10). Therefore, we assume the system lifetime as that when the first node exhausts its battery power. The lifetime of the MS scheme is 5.5 times as long as that of non-scheduling, and 1.85 times as long as that of the PS scheme. A major contributing factor is the effect of reduction of consumption energy by data aggregation. Another possible factor is that sensor nodes turn off their radios shortly after finishing data sending. However, sensor nodes turn their radios off periodically using LPL, so the influence by the second factor is small.

Next, Fig. 11 shows lifetimes when the density of the sensor nodes is changed (case (b)). As in the case (a), even if the density doubles, the MS scheme is effective.

Finally, Fig. 12 shows lifetimes when the maximum number of hops from the base station is changed (case (c)). In this case, the MS scheme is more effective than that in the case (a). Because of data aggregation, the amount of relay data at the sensor nodes near the base station in the MS scheme is fewer than that in the PS Scheme. In addition, the MS scheme can sleep until before the receiving time. The lifetime of the MS scheme is 7.9 times as long as that of non-scheduling, and 2.6 times as long as that of the PS

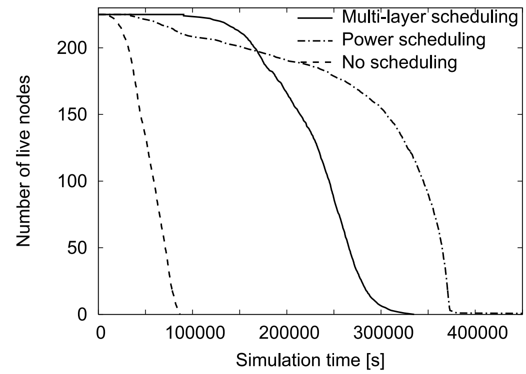


Fig. 12 The lifetime in 225 sensor nodes in a $150\,\mathrm{m} \times 150\,\mathrm{m}$ rectangular area (c).

scheme.

5. Conclusions

In this work, we proposed a scheduling scheme based on RTS/CTS exchange and evaluated our scheme through simulation. Comparison of the results achieved using the proposed scheme with the PS shows a prolonged system lifetime through the use of our scheme. It is important to determine the schedule considering data aggregation.

Acknowledgments

This research work was partially supported by Strategic Information and Communications R&D Promotion Programme (SCOPE), Ministry of Internal Affairs and Communications, Japan, by a Ministry of Education, Culture, Sports, Science and Technology (MEXT) Grant-in-Aid for Scientific Research (C), No. 18500052, and by a Grant-in-Aid from that Ministry for Young Scientists (B), No. 16760271.

References

[1] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," Proc. Hawaii Conference on System Sciences, pp.3005–3014, Jan. 2000.
[2] M.L. Sichitiu, "Cross-layer scheduling for power efficiency in wireless sensor networks," Proc. IEEE INFOCOM 2004, vol.3, pp.1740–1750, March 2004.
[3] S. Lindsey and C.S. Raghavendra, "PEGASIS: Power efficient GAthering in sensor information systems," Proc. IEEE Aerospace Conference, vol.3, pp.1125–1130, March 2001.
[4] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, "Impact of network density on data aggregation in wireless sensor networks," Proc. 22nd International Conference on Distributed Computing Systems, pp.457–458, Nov. 2001.
[5] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed diffusion for wireless sensor networking," Proc. IEEE/ACM Trans. Netw., vol.11, no.1, pp.2–16, Feb. 2003.
[6] S. Siingh, M. Woo, and C.S. Raghavendra, "PAMAS: Power aware multi-access protocol with signalling for ad hoc networks," Proc. ACM/IEEE Conference on Mobile Computing and Networking, vol.28, pp.5–26, Oct. 1998.

[7] J. Heidemann, F. Silva, and D. Estrin, "Matching data dissemination algorithms to application requirements," Proc. ACM SenSys Conference, pp.218–229, Nov. 2003.
[8] P. Karn, "MACA-A new channel access method for packet radio," Proc. 9th Computer Networking Conference, pp.134–140, Sept. 1990.
[9] S. Lindsey, C. Raghavendra, and K. Sivalingam, "Data gathering in sensor networks using the energy*delay metric," Proc. 15th International Parallel & Distributed Processing Symposium (IPDPS-01), pp.2001–2008, April 2001.
[10] B. Krishnamachari, D. Estrinf, and S. Wicker, "Modelling data-centric routing in wireless sensor networks," IEEE INFOCOM, pp.2–14, June 2002.
[11] A. Wang, W.B. Heinzelman, A. Sinha, and A.P. Chandrakasan, "Energy-scalable for battery-operated MicroSensor networks," Kluwer J. VLSI Signal Processing, vol.29, pp.223–237, Nov. 2001.
[12] J. Zhao, R. Govindan, and D. Estrin, "Computing aggregates for monitoring wireless sensor networks," Proc. IEEE International Workshop on Sensor Network Protocols and Applications, pp.139–148, May 2003.
[13] T. Matsuda, M. Ichien, S. Mikami, H. Kawaguchi, C. Ohta, and M. Yoshimoto, "Data transmission scheduling based on RTS/CTS exchange for periodic data gathering sensor networks," Proc. International Conference on Communications and Electronics, CD-ROM, Oct. 2006.
[14] M.L. Sichitiu and C. Veerarittiphan, "Simple, accurate time synchronization for wireless sensor networks," Proc. IEEE Wireless Communications and Networking Conference, vol.2, pp.16–20, March 2003.
[15] K. Romer, "Time synchronization in ad hoc networks," Proc. 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing, pp.173–182, Oct. 2001.
[16] J. Elson and D. Estrin, "Time synchronization for wireless sensor networks," Proc. 2001 International Parallel and Distributed Processing Symposium, Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, pp.1965–1970, April 2001.
[17] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," UCLA Technical Report 020008, Feb. 2002.

Takashi Matsuda received a B.E. degree in Computer and Systems Engineering in 2005 and a M.E. degree in Science and Technology both from Kobe University. He is currently a Doctor's student at Kobe University. His interests include routing protocol for sensor networks.

Masumi Ichien received a B.E. degree in Computer and Systems Engineering from Kobe University in 2005. He is currently a Master's student at Kobe University. His interests include media access controls for sensor networks.

Hiroshi Kawaguchi received the B.E. and M.E. degrees in electronic engineering from Chiba University, Chiba, Japan, in 1991 and 1993, respectively, and the Ph.D. degree in engineering from the University of Tokyo, Tokyo, Japan, in 2006. He joined Konami Corporation, Kobe, Japan, in 1993, where he developed arcade entertainment systems. He moved to the Institute of Industrial Science, the University of Tokyo, as a Technical Associate in 1996, and was appointed a Research Associate in 2003. In 2005, he move to the Department of Computer and Systems Engineering, Kobe University, Kobe, Japan, as a Research Associate. Since 2007, he has been an Associate Professor with the Department of Computer Science and Systems Engineering, Kobe University. He is also a Collaborative Researcher with the Institute of Industrial Science, the University of Tokyo. His current research interests include low-power VLSI design, hardware design for wireless sensor network, and recognition processor. Dr. Kawaguchi was a recipient of the IEEE ISSCC 2004 Takuo Sugano Outstanding Paper Award and the IEEE Kansai Section 2006 Gold Award. He has served as a Program Committee Member for IEEE Symposium on Low-Power and High-Speed Chips (COOL Chips), and as a Guest Associate Editor of IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences. He is a member of the IEEE and ACM.

Chikara Ohta was born in Osaka, Japan, on July 25, 1967. He received the B.E., M.E. and Ph.D. (Eng.) degrees in communication engineering from Osaka University, Osaka, Japan, in 1990, 1992 and 1995, respectively. From April 1995, he was with the Department of Computer Science, Faculty of Engineering, Gunma University, Gunma, Japan, as an Assistant Professor. In October 1996, he joined the Department of Information Science and Intelligent Systems, Faculty of Engineering, University of Tokushima, Tokushima, Japan, as a Lecture, and there he had been an Associate Professor since March 2001. Since November 2002, he has been an Associate Professor of the Department of Computer and Systems Engineering, Faculty of Engineering, Kobe University, Japan. From March 2003 to Feburary 2004, he was a visiting scholar in the University of Massachusetts at Amherst, USA. His current research interests include performance evaluation of communication networks. He is a member of IPSJ, IEEE and SIGCOMM.

Masahiko Yoshimoto received a B.S. degree in electronic engineering from Nagoya Institute of Technology, Nagoya, Japan, in 1975, and an M.S. degree in electronic engineering from Nagoya University, Nagoya, Japan, in 1977. He received a Ph.D. degree in Electrical Engineering from Nagoya University, Nagoya, Japan in 1998. He joined the LSI Laboratory, Mitsubishi Electric Corp., Itami, Japan, in April 1977. From 1978 to 1983 he was engaged in the design of NMOS and CMOS static RAM including a 64 K full CMOS RAM with the world's first divided word-line structure. From 1984, he was involved in research and development of multimedia ULSI systems for digital broadcasting and digital communication systems based on MPEG2 and MPEG4 Codec LSI core technology. Since 2000, he has been a Professor of the Dept. of Electrical and Electronic Systems Engineering at Kanazawa University, Japan. Since 2004, he has been a Professor of the Dept. of Computer and Systems Engineering at Kobe University, Japan. His current activity is focused on research and development of multimedia and ubiquitous media VLSI systems including an ultra-low-power image compression processor and a low power wireless interface circuit. He holds 70 registered patents. He served on the Program Committee of the IEEE International Solid State Circuit Conference from 1991 to 1993. In addition, he has served as a Guest Editor for special issues on Low-Power System LSI, IP, and Related Technologies of IEICE Transactions in 2004. He received the R&D100 awards from R&D Magazine for development of the DISP and development of a real-time MPEG2 video encoder chipset in 1990 and 1996, respectively.