Implementation and Evaluation of a Synchronization-based Data Gathering Scheme for Sensor Networks

Shuntaro Kashihara, Naoki Wakamiya, Masayuki Murata
Graduate School of Information Science and Technology Osaka University
1–5 Yamadaoka, Suita, Osaka 565–0871, Japan
tel: +81-6-6879-4542
s-kasihr@ist.osaka-u.ac.jp, wakamiya@ist.osaka-u.ac.jp, murata@ist.osaka-u.ac.jp

***Abstract*-** **One can obtain information about a region by deploying a network of sensor nodes there. Since remotely deployed nodes are usually powered by batteries, an energy-efficient data gathering scheme is needed to prolong the lifetime of the sensor network. We proposed a novel scheme for periodic data gathering but evaluated in only in simulation experiments assuming ideal environments. In this paper, we evaluated the scheme experimentally in small networks consisting of commercial, off-the-shelf wireless sensor units. We also developed mechanisms to solve problems due to the instability of radio communications and demonstrated the effectiveness of these mechanisms experimentally. We confirmed that energy-efficient data gathering can be implemented by using our proposed scheme with several improvements and that synchronization can be established and maintained under unstable and changing conditions.**

***Keywords*-Sensor Network, Data Gathering, Synchronization, Pulse-Coupled Oscillator Model, MOTE**

I. Introduction

The development of low-cost microsensor equipment having the capability of wireless communications [1] has caused sensor network technology to attract the attention of many researchers and developers. One can obtain information on behavior, condition, and position of elements in a region by deploying a network of battery-powered sensors there. Each node in such a sensor network is equipped with one or more sensors, a general purpose processor with a limited computational capacity, a small amount of memory, and a radio transceiver. The data gathered by the sensors is transmitted directly or indirectly to a base station and provided to users there or accessing the base station through the Internet.

Since a sensor node is typically powered by a battery that cannot be replaced often, the network must use a data gathering scheme that is energy-efficient. The scheme must also adapt to the addition, removal, and movement of sensor nodes automatically without any manual operations of users or administrators. In addition, because sensor nodes are often deployed and distributed in an uncontrolled way, the data gathering scheme cannot be a centralized scheme with a single node or server maintaining all the information and having all the control functions. Data gathering schemes such as LEACH [2] the chain-based protocol [3], and CMLDA [4] cannot function without such global information as the number of sensor nodes deployed, their locations, the predetermined optimal number of clusters, and the residual energy of all sensor nodes. They therefore need additional, possibly expensive and unscalable, communication protocols for collecting and sharing the global information, and they cannot easily adapt to the addition, removal, or movement of sensor nodes.

We previously proposed a scalable, robust, and energy-efficient scheme for periodic data gathering in sensor networks [5]. It can be scaled to the number of sensor nodes and the extent of the monitored region, withstands the failure of sensor nodes, adapts to the addition, removal, and movement of sensor nodes, consumes little power, and does not use a centralized control mechanism.

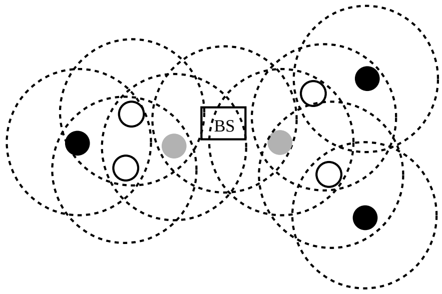

Figure 1: Synchronization-based data gathering

Our scheme transfers information from all the sensor nodes to a base station periodically and saves power by not sending unnecessary data and turning off the data-transmitting components whenever they are not needed. In our scheme, sensor information periodically propagates from the edge of a sensor network to a base station as the propagation forms a concentric circle without any centralized control. In the sensor network illustrated in Fig. 1, for example, the sensor nodes denoted by filled circles, which are the most distant from the base station, transmit sensor information in synchrony. The sensor information is broadcasted and propagates to the range of radio signal depicted as a dashed circle. Sensor nodes (open circles) that are in the range of radio signals but closer to the base station receive the transmitted information, aggregate it with their own sensor information by some data fusion algorithm [6], and broadcast it at the same instant. This information is received by the innermost sensor nodes (grey circles), combined with the information gathered by the sensors at those nodes, and relayed to the base station. Consequently, sensor information from all sensor nodes is gathered at the base station. Those sensor nodes on the same circumference periodically broadcast their sensor information at the same time, slightly before the sensor nodes on the inner circumference broadcast their information. In this synchronized data gathering, each sensor node needs to turn on its transceiver component only at regular intervals, and the amount of data that needs to be transferred can be effectively reduced as illustrated in Fig. 2.

To implement this synchronized data gathering without any centralized controls, we need to have each sensor node independently determines when it should receive and transmit information by observing the radio signals sent by sensor nodes in its vicinity. A clock synchronization method [7] is helpful to some extent, but it consumes energy for clocks with much skew of widely deployed sensor nodes to keep synchronized. We there-

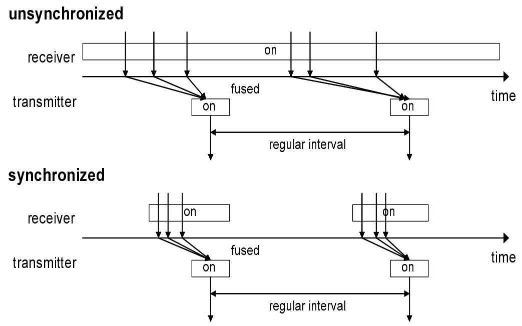

Figure 2: Power saving in synchronization-based data gathering

fore use a pulse-coupled oscillator model [8, 9] based on biological mutual synchronization such as that observed in flashing fireflies, chirping crickets, and pacemaker cells. Our scheme requires no additional signaling mechanism for synchronization: sensor nodes simply transmit sensor information at their own intervals. We confirmed through simulation experiments that our scheme can gather sensor information in a fully-distributed, self-organizing, robust, adaptable, scalable, and energy-efficient manner.

Our simulation evaluation, however, assumed ideal environments in which there was no collision in the medium access layer, all wireless communications were bidirectional and stable, and there was no propagation delay. In real environments, the collision of radio signals causes message to be delayed or even lost, and the radio signal from a sensor node does not always arrive at another node even if that node is within the radio signal propagation range of the sending node. The range of radio signal often changes because of interference among radio signals and because of the reflections and disturbances due to obstacles such as walls, floors, ceilings, and human beings.

In the work reported in this paper, we evaluated the feasibility of our scheme experimentally by using it in sensor networks composed of commercial and off-the-shelf wireless sensor units: MOTE [10]. Because we then found that our scheme could not provide synchronized data gathering because of unstable and unidirectional wireless communications, we developed ways to solve these problems. We then confirmed the energy-efficient synchronization-based data gathering can be obtained using proposed scheme with several improvements under actual conditions. We also confirmed that our scheme adapts to the addition and removal of sensor nodes and to changes in the frequency of data gathering.

The rest of this paper is organized as follows. In Section II., we explain the outline of our synchronization-based data gathering scheme, and in Section III., we describe the sensor network we implemented. In Section IV., we report our experimental evaluation of our scheme, and in Section V., we proposed several improvements. Finally, in Section VI., we conclude by briefly summarizing the paper and discussing our plans for future work.

II. Synchronization-based Data Gathering Scheme

In our synchronization-based data gathering scheme, the timing with which a sensor node broadcasts its sensor information is determined in accordance with its distance from the base station– that is, with what we call its level. In our synchronization-based data gathering scheme, the sensor information of distant sensor nodes is relayed to the base station by closer sensor nodes. Our scheme thus uses multihop transmission and level corresponds to the number of hops from the base station. The level of sensor nodes expressed by the grey circles in Fig. 1, for example, is one, and these nodes can receive the radio signals of the base station directly. Sensor nodes expressed by open circles are on level 2 and can communicate with sensor nodes on level 1 by exchanging radio signals. We assume that a sensor node can aggregate its local sensor information with the sensor information it receives from the other sensor nodes. A message from a sensor node on level 1 thus carries not only its own sensor information but also sensor information obtained at sensor nodes on level 2 or higher.

Periodic data gathering is efficient in terms of power consumption when sensor nodes on the same level synchronously send their information to sensor nodes one level lower (i.e., nodes on a smaller, closer-to-the-base-station, circle). In addition, since each sensor nodes sends its sensor information at its own timing, without waiting for the reception of sensor information from other nodes, sensor nodes must send their information slightly before the nodes at the next lower level send theirs. Therefore, in our scheme, if the base station needs sensor information at time t, sensor nodes on level 1 simultaneously emit their information at $t-\delta$, where δ is large enough to allow for delays cause by collisions and deferment of signal transmission in the medium access layer. Correspondingly, all the sensor nodes on level 2 should send their information at $t-2\delta$.

The timing with which each sensor node sends its information in synchrony with the other nodes on the same level must be determined by each node independently and autonomously. And for energy-efficiency, this synchronization should be accomplished without exchanging any additional control messages. We based our synchronization method on a pulse-coupled oscillator model [8, 9]. Consider a set of N oscillators O_i. Each oscillator has a phase $\phi_i \in [0,1]$ and a state $x_i \in [0,1]$, which is derived by a monotonically increasing function f_i as $x_i = f_i(\phi_i)$. As time passes, ϕ_i shifts toward one and, after reaching it, jumps back to zero. When x_i reaches one, O_i fires and x_i is initialized to zero. The fire stimulates other oscillators O_j as $x_j(t^+) = B(x_j(t) + \epsilon(\phi_j))$. Function B is defines as

$$B(x) = \begin{cases} x, & \text{if} \quad 0 \le x \le 1 \\ 0, & \text{if} \quad x < 0 \\ 1, & \text{if} \quad x > 1 \end{cases} .$$

When x_j reaches one, O_j also fires. O_i and O_j are then synchronized. In our scheme, we consider a set $\mathcal{S}$ of N sensor nodes: $\mathcal{S} = \{S_1, \cdots, S_N\}$. Node S_i is on level l_i, which is initially set to infinity or a reasonably large value, and has a timer and a state. A state $x_i \in [0,1]$ is given by smooth and monotonically increasing function $f_i : [0,1] \rightarrow [0,1]$ of a phase $\phi_i \in [0,1]$ of the timer. For example, we used the following f_i [8]:

$$\forall i, f_i(\phi_i) = \frac{1}{b}\ln[1 + (e^b - 1)\phi_i], \tag{1}$$

where $b > 0$ is one of the parameters that determines the rate of synchronization and is called the dissipation. As the dissipation b increases, f_i raises more rapidly and, as a result, synchrony emerges faster. The base station sends beacon signals at a regular interval to make sensor nodes within the range of its radio signal synchronize with each other. It broadcasts a message specifying level 0. Consider that sensor node S_i receives a message from sensor node S_j at time t. When the level l_j of node S_j is smaller than the level l_i of node S_i, node S_i first changes its level to $l_j + 1$. It is then stimulated and its state changes as follows:

$$x_i(t^+) = B(x_i(t) + \epsilon). \tag{2}$$

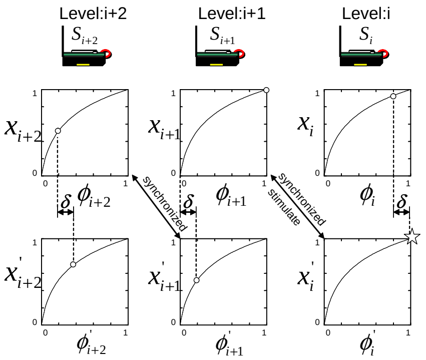

Figure 3: Relationship among state x, phase ϕ, regulated state x', and regulated phase ϕ' of sensor nodes

When node S_i is stimulated and state x_i is raised to one, node S_j's message emission and node S_i's timer are considered synchronized and state x_i and phase ϕ_i jump back to zero.

In the proposed scheme, the timer of node S_i synchronizes with radio signals from node S_j ($l_j = l_i - 1$). Therefore, as explained previously, node S_i must send its sensor information slightly (by offset δ_i) before state x_i becomes 1. Node S_i's offset δ_i can be the same as that of all the other nodes in the network or it can be specified by node S_j in accordance with the number of nodes arround S_j. To take into account offset δ_i, we consider a regulated phase ϕ'_i given by the following equation:

$$\phi'_i = p(\phi_i, \delta_i) = \begin{cases} \phi_i + \delta_i, & \text{if } \phi_i + \delta_i \le 1 \\ \phi_i + \delta_i - 1, & \text{otherwise} \end{cases} . \quad (3)$$

From ϕ'_i, we obtain a regulated state x'_i by $f_i(\phi'_i)$. The regulated state x'_i of stimulated node S_i is given as

$$x'_i = f_i(p(g_i(x_i(t^+)), \delta_i)), \quad (4)$$

where $g_i = f_i^{-1}$. Node S_i sends a message when its regulated state x'_i becomes one. A message that a node S_i sends contains the identifier S_i, the level l_i, and the sensor information of that node, possibly aggregated with other sensor information. If needed, a node's message can also carry a stimulus for the nearby nodes and the offset needed for its child nodes to use the same timing.

When the level l_j of node S_j is larger than level l_i of node S_i by one, node S_i deposits the received sensor information in its local buffer. Sensor information is aggregated to reduce the amount of information and save power. In [2], for example, data fusion results in $2n$ or more bits of sensor information being aggregated to n bits of sensor information.

The level that sensor node S_i belongs to is given as the smallest level l_{min} among the nodes from which node S_i can receive signals plus one, i.e., $l_i = l_{min} + 1$. When a new sensor node occasionally receives a message from a more distant node, it first determines its level wrongly. As time passes, however, it receives another signal from a node closer to the base station. Eventually it identifies its level correctly. Since a node is not stimulated by a message from a node whose level is the same or higher, there is no direct interaction among sensor nodes on the same level. The intra-level synchronization is attained through inter-level stimulation.

As an example, Fig. 3 illustlates the relationship among state x, phase ϕ, regulated state x', and regulated phase ϕ' of sensor nodes. When the regulated state x'_i reaches one, node S_i emits a message. If the state x_{i+1} of the node S_{i+1} reaches one by being stimulated, the regulated state x'_i and the state x_{i+1} get synchronized. By the regulated state x'_{i+1}, which reaches one earlier than the state x_{i+1}, node S_{i+1} emits a message before node S_i's emission by offset δ.

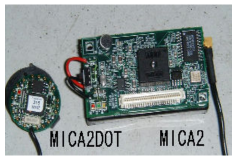

Figure 4: MOTE sensor nodes

III. Implementation of Synchronization-based Data Gathering Scheme

A. Outline of Implemented System

We implemented our synchronization-based data gathering scheme on a sensor network composed of sensor nodes and one computer-controlled base station. We used the commercial sensor unit Crossbow MOTE [10] developed in the NEST (Network Embedded Systems Technology) project of UC Berkeley. It is controlled by TinyOS. As shown in Fig. 4, there are two kinds of MOTE sensor nodes. One is the card-size MICA2, which operates on two AA batteries and can have either of sound, light, temperature, acceleration, and magnetic sensors. The other is the coin-size MICA2DOT, which operates on a button cell battery and has a temperature sensor. Each has a radio transmitter and an omni-directional antenna, and they communicates with each other by using the CSMA/CA MAC protocol and 315MHz/FSK radio signals. The range of the radio signal varies from 5 to 150 m, depending on the transmission power.

The behavior of sensor nodes is controlled by our data gathering scheme. Each sensor node maintains a timer, a state, a level, and sensor information. When the regulated state, which is determined in accordance with the phase of the timer and the offset as specified by (4), reaches one, the node obtains local sensor information and aggregates it with sensor information received from other sensor nodes, if there is any. It then broadcasts a message consisting of its identifier, its level, its sensor information, and other parameters that are described in section B.. When a node receives a message from another node closer to the base station or a beacon signal from the base station, it is stimulated. The frequency with which the base station sends beacon signals is controlled by the computer connected through an RS-232 port, the same port through which the sensor information received by the base station is sent to the computer.

B. Implementaion with MOTE sensors

MOTE sensors have a millisecond clock, and we made a timer with a 10-second cycle by shifting phase ϕ_i by 0.01 every 100 milliseconds. After the phase shift, a sensor node derives state x_i as specified in (1) and obtains the regulated state x'_i from regulated phase ϕ'_i as specified by (4). When state x_i reaches one, state x_i and phase ϕ_i jump back to zero. When the regulated state x'_i reaches one, the node first obtains sensor information from its local sensor and sends the information generated by combining its own sensor information with stored information.

A message is 1232 bits long, and the first 72 bits are information local to the node sending the message. The first 16 bits are the sensor node identifier, and 8 bits indicate the level of the node. Initially, it is set at 255. Sensor information (e.g., monitored temperature) is expressed with a precision of 16 bits, and the next 16 bits are assigned to the local timestamp. Offset (8

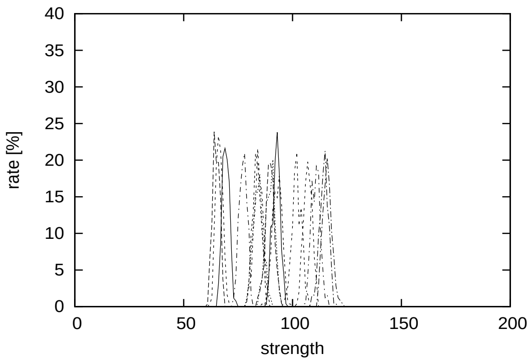

Figure 5: Variation in reception strength among sensor nodes

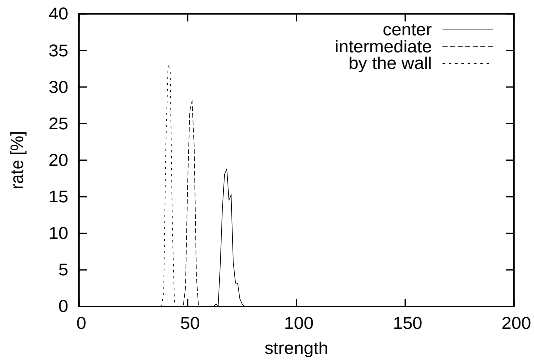

Figure 6: Variation in reception strength on the roof

bits) and stimulus (8 bits) are set when a sender tells receivers to use those values in (2) and (3). The succeeding 40 bits are reserved for debugging purposes. A message can further carry a list of sensor identifiers, levels, sensor information, and timestamps for twenty nodes from which the sending node received sensor information. As explained in section II., our proposed scheme assumes that sensor information will be aggregated to save power. The sensor nodes in our experimental implementation, however, stored and relayed all received sensor information because we wanted to investigate the detailed operation of the scheme. We used the same message format for beacon signals.

The range of radio signal with the minimum transmission power of -20 dBm ranges from 5 to 10 m. To evaluate a sensor network with a radius of four hops, we need a region 80 m in diameter and it becomes difficult to observe the condition of all sensor nodes. We therefore decreased the size of the experimental network by virtually limiting the range of the radio signals (i.e., by ignoring radio signals whose reception strength was below a threshold). The relation between the reception strength and the range of radio signal, however, varies because of the heterogeneity of sensor nodes and the influence of the surroundings.

The variation of reception strength observed at the base station is shown in Fig. 5 for 16 sensor nodes 1 m from the base station. Each node sent signals 300 times with the minimum transmission power -20 dbm. The horizontal axis is the reception strength obtained through a TinyOS API, and the vertical axis corresponds to the ratio of the number of obserbations of the value to 300 trials. A smaller value on the horizontal axis means a greater reception strength. As shown in Fig. 5, the reception strength varied from 61 to 123, differing among nodes and among the 300 signals sent by each of the nodes. We next evaluated influences of walls. Figure 6 shows the variation of reception strength when a base station and a sensor node 1 m apart were put at the center of the roof of a building, when they were put close to a wall 1 m high, and when they were put between the wall and the center of the roof. As shown in Fig. 6, because of the reflection, the reception signal became stronger when they were closer to the wall. We also made reception strength measurements when the sensor node and base station were placed at various orientations in a corridor inside the building. The angle against the wall was changed from 0 degrees to 45 degrees and 90 degrees. We found that, reception was obviously influenced by reflection from the wall. We also made a small network of two sensor nodes so that we could examine the effects of the interference of radio signals. We changed the angle from 0 degrees to 45, 90, and 180 degrees, and the results showed that the radio signals interfered with each other. A radio signal was considerably weakened when the two sensor nodes were arranged nearby. From above experimental results, we decided to empirically control the range of radio signals to between 1.0 and 1.6 m by introducing a threshold and ignoring radio signals whose reception strengths were more than 120.

In conclusion, a sensor node behaves as follows when it receives a radio signal from another sensor node or the base station. It first measures the reception strength and ignores the received message if its reception strength is larger than the predetermined threshold (120). It then compares its own level with the level in the message. If the former is smaller than the latter by more than two, it ignores the message. If the former is smaller than the latter by one, it stores the received sensor information in its local memory. If the number of stored sensor information exceeds 20, the following messages are simply discarded. If its own level is larger than the level in the message, the sensor node is stimulated. It adjusts its level and raises its state x by ϵ. When the regulated state x' reaches one as a result of stimulation, it sends a message that carries aggregated sensor information.

When state x_i of node S_i becomes 1 as a result of being stimulated by node S_j, the stimulating node S_j and the stimulated node S_i are considered synchronized. More specifically, the timer in node S_i is synchronized with the cycle and timing with which node S_j sends messages. Once the synchronization is attained, it is kept as long as the wireless communications are relatively stable. When a node repeatedly receives a stimulating radio signal and its state is always brought to one, it is considered fully synchronized and it goes into a power-saving mode. In this mode the node S_i turns its radio transceiver off from $\phi_i = 0.0$ to $1.0 - 2\delta_i$ and turns it on at $\phi_i = 1.0 - 2\delta_i$ in order to receive sensor information and/or a stimulus and send a message.

IV. Evaluation of Implemented Data Gathering Scheme

A. Evaluation Environment

We conducted empirical experiments on a roof of a building where surroundings were less influential than in the building (see Fig. 7). However, radio conditions were not stable as assumed in our previous work. We arranged the sensor nodes to form circles centered on the base station.

All sensor nodes had a temperature sensor. When the regulated state of a node reached one, the node obtained sensor information and sent a message. The radii of the circles were 1.0, 2.0, and 3.0 m. As described in Section III., we limited the range of radio signals to between 1.0 and 1.6 m by setting

Figure 7: Experimental environment

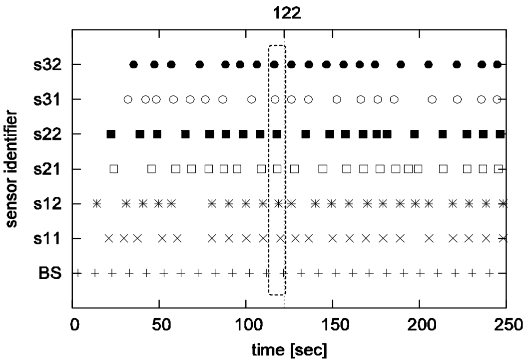

Figure 8: Timing of message emission in the proposed scheme

a threshold reception strength. To obtain as many experimental results as possible in each experiment, we set the cycle of the timers in the nodes to ten seconds and configured the base station to send a beacon signal every ten seconds. All sensor nodes used the same function $f_i(\phi_i)$ in (1) and the same set of parameters: $b = 3.0$, $\epsilon = 0.3$, and $\delta = 0.2$ [5]. Consequently, sensor nodes on the n-th circumference should send their sensor information $2n$ seconds before the base station sends a beacon signal. When all sensor nodes sent their information with a timing appropriate to their locations, global synchronization was considered to have been attained.

B. *Experiment Results*

The results of an experiment of the proposed scheme are shown in Fig. 8. We used a network of six sensor nodes. The first digit following s in the sensor identifier corresponds to a circumference on which the node was located. BS stands for base station, and dots correspond to instants when the sensor nodes and the base station sent messages. Gaps or blanks observed about 70 seconds for nodes s11 and s12 correspond to message loss due to collisions of radio signals. Global synchronization was accomplished at 122 seconds. Nodes s11 and s12, closest to the base station, sent sensor information at the same time, about 2 seconds before the beacon signal. Nodes s21 and s22, farther from the base station, sent sensor information about 4 seconds before the beacon signal. The nodes farthest from the base station, nodes s31 and s32, sent sensor information about 6 seconds before the beacon signal. As a result, sensor information propagated from the edge of the sensor network to the base station under an experimental environment.

The synchronization was immediately lost, however, because of delays caused by collisions and deferment of radio signals. For example, assume that node S_i is synchronized with node S_j and they have a timer with the same cycle. A message sensor node S_j sends should reach node S_i when the state x_i of node S_i reaches one. When node S_j defers message emission on detecting a radio signal from another node to avoid collision and a message arrives at node S_i 0.01 later, the phase of node S_i has already become 0.01. Then node S_i is stimulated, its state and phase change, and the synchronization is lost.

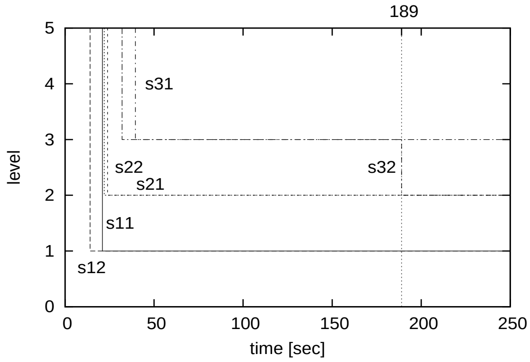

Figure 9: Transition of levels in the proposed scheme

Figure 9 illustrates the transition of the levels of sensor nodes. The levels were initially set to a reasonably large value, 255. It can be seen that the nodes accurately identified their levels by receiving messages from other nodes that had already determined their levels. At 189 seconds, however, the level of node s32 on the third circle became two. It was because it accidentally received from node s11 a radio signal that propagated farther than expected due to the influence of circumstances. Once a sensor node wrongly considers its level smaller as a result of receiving such an anomalous and unstable radio signal, it becomes isolated. Since such signals do not arrive continuously, the node cannot be further stimulated and thus cannot attain synchronization. It is not stimulated by radio signals from nearby nodes, because their levels are the same or larger. In addition, since such communications are asymmetric, its sensor information is not received by the stimulating node. It receives sensor information from sensor nodes whose levels are larger by one but does not relay this information to the base station. Furthermore, radio signals emitted by a sensor node with an incorrectly small level stimulate other neighboring nodes. As a result, the synchronization-based data gathering fails.

V. Improvement of Synchronization-based Data Gathering Scheme

As described in Section IV., the proposed scheme fails in environments where wireless communications are not stable and there are congestions. In this section, we describe three filtering mechanisms we developed to improve the scheme and provide synchronization-based data gathering.

A. *Filtering Mechanisms*

Message transmission is deferred to avoid collisions of radio signals. To keep nodes from being stimulated by such delayed signals, we developed a new filtering mechanism: a sensor node ignores radio signals whose level are smaller by 1 if they arrive slightly after when its state changed from 1 to 0. In the experiments reported in this paper we set this period to 0.6 seconds, taking into account that the maximum back-off time observed was 0.26 seconds.

By being reflected by obstacles such as walls, radio signals sometimes accidentally reach unexpectedly distant sensor nodes.

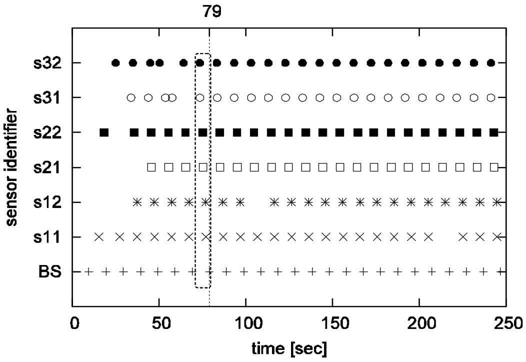

Figure 10: Timing of message emission in the improved scheme

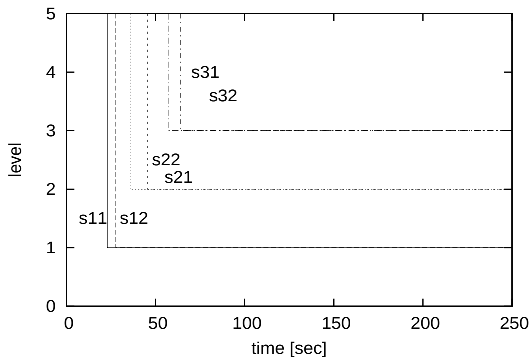

Figure 11: Transition of levels in the improved scheme

Since such radio signals are usually unstable and communications among a sender and a receiver become unidirectional and asymmetric, a sensor node must ignore them and not change its level. We developed two more filters to solve this problem. One is to ignore radio signals with insufficient reception strength (e.g., more than 100 in our experiments). The other is to ignore radio signals that arrive infrequently and intermittently. In our experiments, a sensor node changed its level only if it received radio signals from nodes with a level more than two smaller than its own while its state reaches one three times.

In conclusion, a sensor node that receives a radio signal from a node whose level is smaller than its own level by more than two behaves as follows. It first compares the strength of the received signal with the predetermined threshold value, 100. If it is larger, the node considers the radio signal too weak for that node to be stimulated and ignores it. It then investigates a list of levels. A new level value is added to the list when a sensor node receives a radio signal of that level, and the list is reinitialized every time the state of the node reaches one three times–in other words, when three timer cycles have passed. If the level of received radio signal is on the list, the node considers the wireless communications stable. Otherwise, it discards the message. Finally, if the reception time is not in the period during which radio signals are to be ignored, the node is stimulated and its level, state, and phase change. When the level of received signal is smaller than its own level by one, only the last filter is used.

B. Evaluation of Improved Scheme

The results obtained when evaluating the improved scheme experimentally are shown in Fig. 10 and Fig. 11. In subsequent experiments we used 16 sensor nodes arranged in a circle around the base station but excluded nodes whose sensor information could not be gathered. In Fig. 10, it is shown that the global synchronization was accomplished at 79 seconds. Nodes s11 and s12 sent their sensor information about 2 seconds before the beacon signal, and nodes s21 and s22 sent their sensor information about 2 seconds before than nodes s11 and s12 did so that their information could be relayed to the base station. Furthermore, nodes s31 and s32 sent their sensor information about 2 seconds earlier so it could be aggregated with the information sent by nodes s21 and s22. It can be seen that when the improved scheme was used, the global synchronization was maintained once it was obtained. As shown in Fig 11, however, time required to appropriately adjust levels became longer than it was when the original scheme was used (Fig 9). This is because nodes were stimulated only by radio signals that arrived more than twice within three timer cycles.

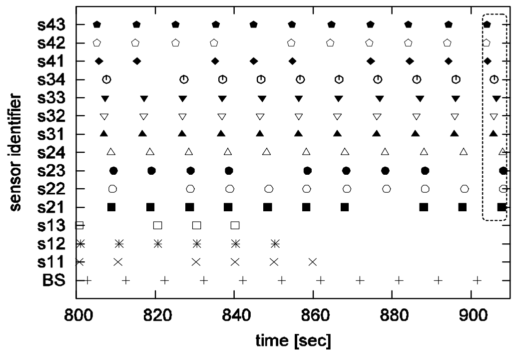

Figure 12: Timing of message emission after removal of sensor nodes

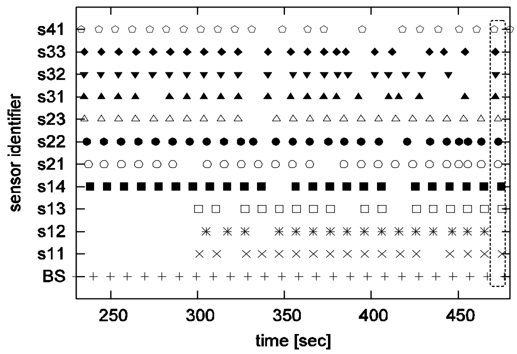

Figure 13: Timing of message emission after addition of sensor nodes

We then evaluated the ability of the network to adapt to the removal of sensor nodes. Figure 12 shows what happened when we removed nodes s13, s12, and s11 at 845 seconds, 855 seconds, and 865 seconds after the global synchronization had been attained at 692 seconds. The figure shows that the global synchronization was maintained even though there were no sensor nodes on the innermost circle and the beacon signals were no longer effective.

We also investigated what happens when sensor nodes are added to the network. Figure 13 shows what we observed when, after the global synchronization had been attained, we added sensor nodes s11, s12, and s13 to the innermost circle at 300 seconds. The levels of those new sensor nodes were initially set

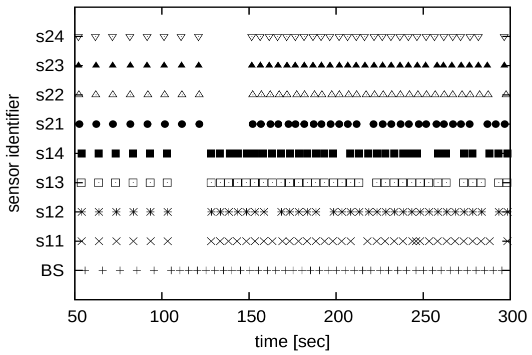

Figure 14: Timing of message emission after change of data gathering frequency

to a sufficiently large value. When they received radio signals–beacon signals in this experiment–twice, they adjusted their level values to one. However, at that time they were not synchronized with the beacon signals, so they sent sensor information at their own independent timings. Since the messages from those nodes were denoted level-1 messages, the level-2 sensor nodes around those new nodes were stimulated and their states were changed. Consequently, the synchronization was lost on the second circle. The influence then propagated outward and all sensor nodes went out of synchronization. After receiving the beacon signals several times, however, the new sensor nodes became synchronized with the beacon signals and began to transmit sensor information in synchrony. The synchronization propagated over the network, and the global synchronization was reestablished at 470 seconds.

We also experimentally confirmed the adaptability of our scheme to changes in the frequency of data gathering. After the global synchronization had been established, we reduced the interval of beacon signals by half at 105 seconds. The results are shown in Fig. 14. Suddenly receiving beacon signals of doubled frequency, the nodes first failed to transmit their sensor information (see the blank periods in Fig. 14). Consider the situation in which node S_i is synchronized with node S_j. When node S_i receives a radio signal from node S_j, the phase ϕ_i of node S_i is one. When the frequency of message transmission from node S_j is doubled, phase ϕ_i is only 0.5 when the signal from node S_j is received. The regulated phase ϕ'_i at this time is $0.5+\delta_i$, i.e., 0.7. Node S_i is stimulated and its state and phase jump to 1.0. As a result, regulated phase ϕ'_i also jumps from 0.7 to 0.2. This means that node S_i loses a chance to transmit its sensor information even though it is synchronized with node S_j. In our scheme, node adjusts the offset as $S_i \rightarrow 1.0 - \phi_i + \delta_i$ in such a case [5]. In the above example, the offset becomes 0.7 and the node can transmit sensor information at phase $\phi_i = 0.3$. In our experiment, all sensor nodes began to transmit sensor information at the doubled frequency at 155 seconds.

VI. Conclusion

In this paper, we first evaluated our synchronization-based data gathering scheme experimentally in a sensor network consisting of commercial, off-the-shelf wireless sensor units. Since we found that the scheme could not establish the synchronization-based data gathering when the wireless communications were unstable, we proposed filtering mechanisms and found that our thus-improved scheme could periodically gather sensor information from sensor nodes and adapt to changes in sensor networks.

The experiments reported in this paper were conducted on a roof with few obstructions, and we are now carrying out further experiments in an environment with more obstructions, more interference, and more collisions, as there are in a building. In addition, we only confirmed that our scheme can adapt to the addition and removal of sensor nodes and to changes of data gathering frequency. We need also to experimentally demonstrate its scalability, robustness, and energy-efficiency.

Acknowledgement

This work was partly supported by "The 21st Century Center of Excellence Program" and a Grant-in-Aid for Scientific Research (A)(2) 16200003 from The Japanese Ministry of Education, Culture, Sports, Science and Technology.

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Computer Networks (Elsevier) journal*, vol. 38, pp. 393–422, Mar. 2002.

[2] W. R. Heinzelman, A. Chandraksan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the IEEE Wireless Communication and Networking Conference*, pp. 3005–3014, Jan. 2000.

[3] S. Lindsey, C. Raghavendra, and K. Sivalingam, "Data gathering in sensor networks using the energy*delay metric," in *Proceedings of the 15th International Parallel & Distributed Processing Symposium (IPDPS-01)*, pp. 2001–2008, Apr. 2001.

[4] K. Dasgupta, K. Kalpakis, and P. Namjoshi, "An efficient clustering-based heuristic for data gathering and aggregation in sensor networks," in *Proceegings of the IEEE Wireless Communication and Networking Conference (WCNC)*, pp. 16–20, Mar. 2003.

[5] N. Wakamiya and M. Murata, "Scalable and robust scheme for data gathering in sensor networks," in *Proceedings of International Workshop on Biologically Inspired Approaches to Advanced Information Technology (Bio-ADIT)*, LNCS3141, pp. 305–314, Jan. 2004.

[6] I. Goodman, R. Mahler, H. Nguyen, I. Goodman, *Mathematics of Data Fusion*, vol. 37 of *Theory and Decision Library. Series B, Mathematical and Statistical Methods.* Norwell, MA, USA: Kluwer Academic Print on Demand, Jan. 1997. ISBN: 0792346742.

[7] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks: A survey," *IEEE Network*, vol. 18, pp. 45–50, July 2004.

[8] R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *Society for Industrial and Applied Mathematics Journal on Applied Mathematics*, vol. 50, pp. 1645–1662, Dec. 1990.

[9] X. Guardiola, A. Diaz-Guilera, M. Llas, and C. Perez, "Synchronization, diversity, and topology of networks of integrate and fire oscillators," *The America Physical Society Physical Review E*, vol. 62, pp. 5565–5569, Oct. 2000.

[10] "MOTE." available at `URL: <http://www.xbow.com/Products/Wireless_Sensor_Networks.htm>`.