A Self-Organizing Communication Mechanism using Traveling Wave Phenomena for Wireless Sensor Networks

Yoshiaki Taniguchi, Naoki Wakamiya, Masayuki Murata
Graduate School of Information Science and Technology, Osaka University
1-5 Yamadaoka, Suita, Osaka 565-0871, Japan
{y-tanigu, wakamiya, murata}@ist.osaka-u.ac.jp

Abstract

In this paper, we propose a self-organizing communication mechanism for a wireless sensor network where a large number of sensor nodes are deployed. To accomplish application-oriented periodic communication without any centralized controls, we adopt traveling wave phenomena of a pulse-coupled oscillator model by regarding sensor nodes as oscillators and the emission of radio signals as firing. We first investigate conditions of a phase-response curve to attain wave-formed firing patterns regardless of the initial phase of oscillators. We adopt the derived phase-response curve to accomplish the desired form of message propagation through local and mutual interactions among neighboring sensor nodes. Through simulation experiments, we confirm that our mechanism can gather or diffuse information effectively in accordance with the application’s requirements.

Keywords—sensor network, traveling wave, pulse-coupled oscillator model

1. Introduction

The development of low-cost microsensor equipment having the capability of wireless communication has caused sensor network technology to attract the attention of many researchers and developers. It is possible to obtain information on behavior, condition, and position of elements in a region by deploying a network of battery-powered sensor nodes there. Each sensor node in such a sensor network has a general purpose processor with a limited computational capacity, a small memory, and a radio transceiver. Since a sensor node is typically powered by a battery that can not be easily replaced, a sensor network must use a communication mechanism that is energy-efficient. In addition, because sensor nodes are often deployed and distributed in an uncontrolled way, the communication mechanism can not be centralized with a single node or a server maintaining all information and having all control functions. Furthermore, according to the application’s requirements, various types of message propagation involving the whole network are required. For example, we need a form of communication where information propagates from the edge of a wireless sensor network to the center to gather information to a base station or a sensor node. We also need another form of communication where information propagates from the center to the edge to distribute information or control signal from a base station or a sensor node.

Most of communication schemes for data gathering cannot adopt to dynamically changing application’s requirements and surrounding condition. For example, directed Diffusion [1] considers both types of communication, i.e., pull and push. The pull-type communication mechanism, where sinks find sources of information of interests by flooding query messages, is shown to be appropriate for a case with many sources and few sinks. The push-type mechanism is, in contrast, good for a case with many sinks and few sources. However, they cannot be used simultaneously and the mechanism to employ must be determined in advance taking into account expected conditions, including the number of sources and sinks and their communication frequency, in the directed diffusion. To answer dynamically changing application’s requirements, the mechanism should handle both types of communication, especially in an autonomous and self-organizing manner. In addition, taking into account the insufficient computational capability and memory capacity of inexpensive small sensor nodes, the mechanism must be as simple as possible. A simple mechanism can also avoid introducing programming and operational errors.

In this paper, we propose a new communication mechanism which can organize a variety of communication, i.e., diffusion and gathering, depending on an application’s requirements. The desired pattern of message propagation emerges through reactions of sensor nodes to the surrounding conditions and local and mutual interactions among sen-
A pulse-coupled oscillator (PCO) model is adopted to make sensor nodes operate in synchrony, e.g., clock synchronization, through a distributed and self-organizing control mechanism [4–8]. In [7, 8], we proposed a data gathering scheme which employs a PCO model to make sensor nodes operate in synchrony, i.e., by self-organization. For this purpose, we adopt a pulse-coupled oscillator (PCO) model based on biological mutual synchronization such as that observed in flashing fireflies [2, 3].

In a PCO model, each oscillator operates on a timer. When a timer reaches one, the oscillator fires. Oscillators coupled with the firing oscillator are stimulated and they shift their timers by a small amount. Through mutual interactions by stimuli among oscillators, they eventually reach a synchronized behavior. There are several papers which employ a PCO model to make sensor nodes operate in synchrony, e.g., clock synchronization, through a distributed and self-organizing control mechanism [4–8]. In [7, 8], we proposed a data gathering scheme which employs synchronized behavior of a PCO model, and confirmed that it worked in a fully-distributed, self-organizing, robust, adaptive, scalable, and energy-efficient manner. In this paper, in contrast to the previous work, we focus on another phenomenon observed in a PCO model. In a PCO model, it is shown that not only the global synchronization where all oscillators fire synchronously, but a traveling wave, where oscillators behave synchronously but with fixed phase difference, appears (Fig. 1) [3]. By adjusting parameters and functions in the PCO model, we can control the frequency, form, and direction of a wave.

In our scheme, each sensor node broadcasts its sensor information in accordance with the phase of its own timer. When a sensor node receives a radio signal of others, it shifts the phase of its timer. Through mutual interactions among neighboring sensor nodes, they reach the state, called phase-lock, where the phase differences among sensor nodes are kept constant, and they emit sensor information alternately. In our previous research work [9], we proposed a PCO-based scheme which could generate various traveling waves, i.e., line, concentric circle, wedge, and radar. However, our scheme was sensitive to the initial phase setting. When the initial phase of sensor nodes was set incorrectly by some reasons such as initialization error or geomagnetism, our scheme could not generate the preferred wave or reach the stable condition. In this paper, we eliminate the requirement on the initial phase. We investigate conditions of a phase response curve (PRC) of the PCO model, with which a wireless sensor network reaches a preferred phase-lock condition from arbitrary settings of the initial phase of sensor nodes. Then, we propose a self-organizing communication mechanism which generates concentric traveling waves centered at a sensor node, which wants to gather information from all sensor nodes or diffuse information to all sensor nodes.

The rest of this paper is organized as follows. First, in Section 2, we briefly introduce the pulse-coupled oscillator model we adopted in this paper. Next, we investigate conditions of a PRC which leads to a desired form of a traveling wave from arbitrary settings of the initial phase in Section 3. Then, we propose a communication mechanism for wireless sensor networks in Section 4, and show simulation results in Section 5. Finally, we conclude the paper and describe future research work in Section 6.

2. Pulse-Coupled Oscillator Model

A pulse-coupled oscillator model is developed to explain synchronous behaviors of biological oscillators such as pacemaker cells, fireflies, and neurons. In this section, mainly following the model described in [3], we give a brief explanation of the model.

Consider a set of $N$ oscillators. Each oscillator $i$ has phase $\phi_i$ ($d\phi_i/dt = 1$). As time passes, $\phi_i$ shifts toward one and, after reaching it, the oscillator fires and the phase jumps back to zero. Oscillator $j$ coupled with the firing oscillator $i$ is stimulated and advances its phase by an amount $\Delta(\phi_j)$. Thus, we have

$$\phi_j \rightarrow \phi_j + \Delta(\phi_j),$$

where $\Delta(\phi)$ is called a phase-response curve (PRC). For example, for the quadratic integrate-and-fire (QIF) model, $\Delta_{QIF}(\phi) = -a \sin 2\pi \phi$ and for the radial isochron clock (RIC) model, $\Delta_{RIC}(\phi) = a(1 - \cos 2\pi \phi)$ [3]. Here, an oscillator ignores all stimuli at the moment of firing [3], and an oscillator identifies multiple stimuli received at the same time as one stimulus.

Through mutual interactions, a set of oscillators reach either of the global synchronization where they have the same phase and fire all at once (see Fig. 2), or the phase-lock condition where phases are different among oscillators with a constant offset (Fig. 3) and the propagation of firings.
seems like a traveling wave. Whether a network reaches the global synchronization or the phase-lock depends on the initial phase of timers or the properties of the PRC [10]. In Fig. 4, $h(\phi)$ indicates the phase at which an oscillator is stimulated again by a neighboring oscillator, after the oscillator is stimulated from a neighboring oscillator at the phase of $\phi$. For example, in the case of a pair of oscillators, it is defined as $h(\phi) = 1 - F(1 - F(\phi))$ where $F(\phi) = \phi + \Delta(\phi)$. A dashed line stands for $h(\phi) = \phi$, and a stepwise line stands for phase transition of an oscillator whose initial phase is $\phi_1$. When an oscillator is stimulated at $\phi_1$, the oscillator changes its phase by using Eq. (1), and it will observe the next fire and be stimulated at $\phi_2$. In Fig. 4, through being stimulated several times, the phase $h(\phi)$ becomes one. It means that an oscillator receives a stimulus from another firing oscillator when the oscillator itself is firing. Therefore, they fire at the same time (Fig. 2). As can be seen in Fig. 4, all oscillators in a system will eventually reach $h(1) = 1$ independently of the initial phase and finally they will fire synchronously. On the contrary, if oscillators have a PRC corresponding to Fig. 5, $h(\phi)$ converges at $\phi_c$ independently of the initial phase. It means that all oscillators reach the condition where the phase is always $\phi_c$ when being stimulated. Therefore, oscillators fire with the time difference of $\phi_c$ at the stable condition.

3. Condition of PRC to Generate Traveling Waves

In this section, we investigate conditions of PRC that lead to desired phase-lock condition regardless of the initial phase to generate preferred traveling waves. We call an oscillator which dominates and controls a PCO network as a pacemaker. To keep the timing and frequency of communication, a pacemaker will not be stimulated and will fire at regular intervals, which corresponds to the data gathering or diffusion cycle in a wireless sensor network.

First, we consider a traveling wave in a PCO network where oscillators are arranged in a line. Oscillators are
numbered by the number of hops from the pacemaker. An oscillator is stimulated only by its neighboring oscillator which is closer to the pacemaker. A pacemaker fires periodically at regular intervals of one time unit. Oscillators fire in order of the pacemaker, oscillator 1, oscillator 2, ⋅⋅⋅, oscillator N at constant phase-difference τ. Therefore, if a pacemaker fires at time 0, oscillator 1 fires at time τ, and oscillator N fires at time Nτ. Here, we consider 0 < τ < 1.

Now, consider phase the transitions of oscillators at the phase-lock condition. Assume that after t time unit since oscillator i (1 ≤ i ≤ N) fired, an oscillator i is stimulated by oscillator i − 1. Oscillator 0 corresponds to the pacemaker. Since oscillators fire at constant phase-difference τ, the phase of an oscillator becomes 1 − τ when it is stimulated by a neighboring oscillator, i.e., \( F(t) = 1 − τ \). Then, oscillator i fires at \( τ + t \). Since an oscillator fires at regular intervals of one at the phase-lock condition, we have \( t + τ = 1 \). Hence, we have

\[
\Delta(1 − τ) = 0. \tag{2}
\]

To generate a desired traveling wave regardless of the initial phase, an oscillator should advance its phase towards \( 1 − τ \) when it is stimulated during \( 0 ≤ φ < 1 − τ \), and push back its phase towards \( 1 − τ \) when it is stimulated during \( 1 − τ < φ < 1 \). Finally, we have following conditions of PRC to generate a traveling wave regardless of the initial phase.

\[
\begin{align*}
0 < \Delta(φ) &\leq 1 − τ − φ \quad (0 ≤ φ < 1 − τ) \\
\Delta(φ) & = 0 \quad (φ = 1 − τ) \\
1 − τ − φ &\leq \Delta(φ) < 0 \quad (1 − τ < φ < 1).
\end{align*} \tag{3}
\]

For example, following PRC functions satisfy Eq. (3).

\[
\begin{align*}
\Delta_1(φ) & = b(1 − τ − φ) \tag{4} \\
\Delta_2(φ) & = a \sin \left( \frac{π}{1 − τ} φ + b(1 − τ − φ) \right) \tag{5}
\end{align*}
\]

Here, \( a (−\frac{b(1−τ)}{b}) < a ≤ 1−\frac{b(1−τ)}{b} \) and \( b (0 < b ≤ 1) \) are parameters which determine characteristics of PRC. Figure 6 illustrates PRC \( \Delta_2(φ) \) for two different settings of \( a \) and \( b \) when \( τ = 0.2 \). Two dot-and-dash lines stand for \( \Delta(φ) = 0 \) and \( Δ(φ) = 1 − τ − φ \), respectively. The curve of PRC satisfying Eq. (3) must lie between these two lines. As parameters \( a \) and \( b \) increase, a traveling wave emerges more rapidly. Similarly, a larger \( b \) of PRC \( \Delta_1(φ) \) leads to the faster emergence of a traveling wave. Especially, a traveling wave emerges by only one interaction, i.e., stimulus, among oscillators with \( b = 1 \). However, such aggressive setting spoils the resilience of the mechanism against a failure of node and unexpected influence from the environment, since a single firing emitted at a wrong time will drastically change the state of the whole system. Therefore, a PRC function and its parameters should be appropriately determined taking into account the trade-off between the speed that a traveling wave emerges and the resilience against failures.

With a PRC satisfying the above conditions, oscillators fire in order of the pacemaker, oscillator 1, oscillator 2, ⋅⋅⋅, oscillator N at constant phase-difference of τ at the phase-lock condition. This can also be regarded as a traveling wave propagating from oscillator N toward the pacemaker, with constant phase-difference \( 1 − τ \). Therefore, to have a diffusion type of communication, where information propagates from the pacemaker to oscillator N with constant phase-difference \( τ \), \( τ \) should be set as \( τ < 0.5 \). On the contrary, to have a gathering type of communication, \( τ \) should be set as \( τ > 0.5 \).

The same PRC can also be applied to the case of two-dimensional arrangement of oscillators. By making a tree whose root is the pacemaker and setting the direction of stimuli as shown in Fig. 7, we can adopt the same PRC and generate a traveling wave propagating from or to the pacemaker in a two-dimensional area. In Fig. 7, each circle stands for an oscillator, and each arrow shows the direction of stimuli. Although any routing protocol for wireless sensor networks is viable to organize such tree-type topology, a simple way of setting such relationship among oscillators will be given in the next section.
4. A Distributed and Self-organizing Communication Mechanism

In this section, we propose a fully-distributed and self-organizing communication mechanism for wireless sensor networks. In our mechanism, any sensor node can gather or diffuse information in accordance with an application’s requirements. For example, a sensor node detecting an emergency would distribute the information over the whole sensor network to alert the other nodes and make them cooperatively cope with the emergency. On the contrary, a sensor node detecting an uncertain condition would collect and aggregate sensor information of the other nodes to have a precise view of the environment by conjecturing from collected information. We define a sensor node which is the center of diffusion or gathering as a core node.

4.1. Basic Behavior

Sensor node \(i\) \((1 \leq \phi \leq N)\) has a timer with phase \(\phi_i \in [0, 1]\). It maintains PRC function \(\Delta(\phi)\), level value \(l_i\), session identifier \(s_i\), parameter \(\delta_i\), and offset \(\tau\) \((0 < \tau < 0.5)\). Initially a level value, which corresponds to the number of hops from the core node, a session identifier, and a parameter are set to zero. The PRC function and offset are determined at the deployment phase, but the offset can be dynamically adjusted taking into account the density of sensor nodes in the whole sensor network. A core node always uses level value of zero, and sets the session identifier at the current value plus one, when it initiates new information gathering or diffusion. In this paper, based on Eq. (5), we use the following PRC function for all sensor nodes.

\[
\Delta(\phi) = a \sin \frac{\pi \phi}{g} + b(\phi - \phi),
\]

Here, \(g\) is defined as \((1 + \delta_i \tau) \mod 1\). \(\delta_i\) is a parameter which controls the direction of information propagation. It is set at 1 for diffusion and -1 for gathering.

A sensor node wakes up when its phase is at \(1 - \tau\), and then it receives and processes messages as needed. When its phase reaches one, a sensor node broadcasts a message. A message that sensor node \(i\) emits, contains the level value \(l_i\), parameter \(\delta_i\), session identifier \(s_i\), and its information aggregated with other sensor’s information which it kept in its buffer. After that, a sensor node keeps awake for \(\tau\) to receive and process messages as needed, and then goes to sleep. \(\tau\) should be appropriately determined considering trade-off between the rate of successful message reception and the lifetime of the sensor network. The smaller \(\tau\) is, the smaller the probability of successful message reception due to possible delay of message transmission caused by collisions in radio signals. At the same time, a smaller \(\tau\) leads to longer lifetime of a sensor network, since a node is awake for the duration of \(2\tau\) in one communication cycle.

Now, sensor node \(i\) receives a message from sensor node \(j\). If session identifier \(s_j\) is larger than \(s_i\), sensor node \(i\) considers that a new communication begins. Therefore, it sets its level value \(l_i\) at \(l_j + 1\), session identifier \(s_i\) at \(s_j\), and parameter \(\delta_i\) at \(\delta_j\). Then, it is stimulated to join a new traveling wave. This mechanism means that the current communication is terminated by the newly initiated communication. To avoid unintended termination of communication by other sensor nodes, a core node might advertise its desired communication period in a message it emits. However, it requires an additional mechanism such as clock synchronization, and it is left as one of future research issues. For the case with multiple core nodes with the same session identifier, details will be given later. If session identifiers are the same among sensor nodes \(i\) and \(j\) but the level value \(l_j\) is smaller than \(l_i\), sensor node \(i\) sets its level value \(l_i\) at \(l_j + 1\), parameter \(\delta_i\) at \(\delta_j\), and it is stimulated. Stimulated sensor node \(i\) adjusts its phase based on the PRC function. As in the PICO model, a sensor node ignores messages from sensor nodes with a smaller level value during the following duration of \(\tau\) when it has already been stimulated to avoid being stimulated by deferred messages for retransmission or collision. If the session identifier is the same and level value \(l_j\) is \(l_j - \delta\), sensor node \(j\) is an upstream node of sensor node \(i\). Therefore, to relay information of sensor node \(j\) to the next downstream node \(i + 1\), sensor node \(i\) deposits the received information in its local buffer. Information aggregation can be done at this time or just before next message emission. If a message does not satisfy the above conditions, sensor node \(i\) ignores it. The algorithm is illustrated in Fig. 8.

4.2. Power-saving Mode

When a sensor node recognizes that a new communication takes place by observing the session identifier, it stays awake to continuously exchange stimuli among neighboring sensor nodes. Through mutual and local interaction among neighboring sensor nodes, the phase-lock condition eventually emerges. Then, a sensor node moves to a power-saving mode by turning off its radio transceiver and other needless modules from \(\phi = \tau\) to \(1 - \tau\). For this purpose, a sensor node has to judge whether the phase-lock condition is globally accomplished or not. When we consider \(T_{\text{max}}\) as the worst-case time required for a sensor node to establish the phase-lock condition with a neighboring node closer to the core node, we can expect that a sensor node can move to a power-saving mode after \(l_i \times T_{\text{max}}\) since the level value is updated. If the phase-lock condition is lost for some reasons after a power-saving mode is activated, a sensor node does not receive any valid message when it is awake. In
such a case, a sensor node stops a power-saving mode to reorganize the phase-lock condition.

4.3. Deployment and Removal of Sensor Nodes

Next, we consider the case where a new sensor node is introduced in a sensor network in operation. Initially, the session identifier of a new sensor node is set at zero. Therefore, it does not affect other sensor nodes. Being stimulated several times, its level value, session identifier, and parameter are correctly identified, and its timer synchronizes at constant phase-difference with that of a neighboring sensor node whose level is smaller by one.

On the contrary, when a sensor node disappears due to battery depletion or removal, a sensor node that is synchronized with the vanished node will be stimulated by another of the same level as the vanishing node. If there is no other node with a smaller level value in its vicinity, the sensor node becomes isolated. Since it does not receive stimuli any more, it can recognize the isolation and then it initializes its session identifier so that it can synchronize with other neighboring sensor nodes.

4.4. Multiple Core Nodes

In addition, we consider the case that there are two or more core nodes with the same session identifier in the sensor network. Since it takes time for information to propagate between the edge of a wireless sensor network and a core node, it is a good idea to have multiple core nodes for one communication to solve the scalability problem. In such case, the sensor network is divided into clusters each of which has one core node. Each core node can gather or diffuse information in its cluster.

A sensor node which is at the same hop count from more than one core node, which we call a border node, receives messages from different clusters. Since session initiation is not necessarily synchronized among core nodes and time required for stimulus propagation would differ among paths from core nodes, such multiple stimuli prevent a border node from establishing the phase-lock condition with neighboring nodes. Therefore, a border node chooses one cluster which it belongs to. First, after a level value is updated, a border node waits for the duration of $(l_i - 1) \times T_{\text{max}} + \tau$ until the phase-lock condition is established in each of clusters. Then, it begins to stick on the timing of the first message it receives. To avoid being stimulated by deferred messages or message originated from another core node, it ignores received stimuli during the following duration of $1 - \tau$ when it has already been stimulated.

4.5. Node Failures

Finally, we consider cases of node failures. First, the failure of a radio transmitter has no influence on other communication, since a failed sensor node can not emit any message and never stimulate neighboring nodes. On the other hand, a sensor node with a failed receiver keeps sending messages based on its timer. The timer of a failed node keeps its own pace independently of the others. Therefore, when the phase-lock condition is not established yet, the failed node disturbs establishment of the phase-lock condition by stimulating neighboring nodes at inappropriate timings. However, a failed node eventually considers it is isolated for not receiving any message from neighboring nodes. Then, it initializes its state and it does not affect others anymore.

In some cases, a timer gains or loses, being affected by, for example, geomagnetism. Basically, a wrong timer will be correctly adjusted from stimulations. A sensor node with a timer which gains, stimulates neighboring nodes at a wrong timing, since sensor nodes take the first message it receives and ignores the following delayed messages. If a wrong timer keeps an advanced phase, the problem is that the interval between message emission of the failed node and its upstream node becomes smaller than $\tau$ and it does
not bring any serious influence on message propagation. On the other hand, a sensor node with a timer which loses does not affect the phase-lock condition very much. If a sensor node which is stimulated by a failed node has another normal node with a smaller level value, it is always stimulated by the normal sensor node and ignores delayed messages from the failed node. Otherwise, the interval of message emission of the failed sensor node and the affected sensor node becomes longer than $\tau$.

In addition, we consider wrong setting of parameters such as $\delta_i$, $l_i$, and $s_i$ of sensor node $i$ by temporal error of memory or CPU. Parameter $\delta_i$ is updated periodically when a sensor node receives a message from a sensor node with a smaller level value. When level value $l_i$ is incorrectly larger than the actual hop count from a core node, messages emitted by sensor node $i$ do not affect neighboring sensor nodes with a smaller level value. On the contrary, if the level value is too small, neighboring nodes would wrongly identify their distance from the core node. It first disturbs establishing the phase-lock condition. However, since the level value of the failed sensor node is the smallest in its range of radio signals, it does not receive any stimulus, i.e., a message with a further larger level value. Therefore, it considers it is isolated and initializes its session identifier so that its level value will be adjusted correctly. When the session identifier $s_i$ is incorrectly smaller than the current session identifier used in a wireless sensor network, the failed sensor node does not affect the others at all. Its session identifier will be corrected on receiving a message from a neighboring sensor node. On the contrary, a larger session identifier $s_i$ means that a new communication is initiated by the failed sensor node. Since the other nodes cannot judge whether a new communication is actually initiated or not, it is handled as normal. When another sensor node initiates a new communication, a new session identifier is used and the failed node does not affect the others any more.

5. Simulation Experiments

We confirm the basic behavior of our communication mechanism through simulation experiments. We consider sensor networks of 100 sensor nodes randomly distributed in a $10 \times 10$ region. The range of the radio signal is fixed at 2 units of length. The initial phase of sensor node is randomly chosen. We use Eq. (6) with $a = 0.01$ and $b = 0.5$ as the PRC function and $\tau$ is set at 0.1. From 0 to 20 time units, we randomly chose a sensor node A as a core node for information diffusion. Then, from 20 to 40 time units, we randomly chose another sensor node B as a core node for information gathering.

Figure 9 shows how the sensor network reached the phase-lock condition in a certain simulation experiment. Each mark stands for an instant when a sensor node emitted a message. For easier understanding, sensor nodes are sorted in order of the hop count from the core node. In the upper figure, at first, all sensor nodes randomly and independently emit messages. However, by exchanging stimuli several times, the phase-lock condition is eventually accomplished and a regular pattern appears. It is clearly shown that message emission is in order of the hop count of sensor nodes, from the core node to nodes with larger numbers. In the lower figure, it is shown that the phase-lock condition for information diffusion is first broken for information gathering initiated by sensor node B. Then, although it takes longer time than for diffusion, the new phase-lock condition appears, where information propagates from the edge of the sensor network towards the core node B.

Over 100 experiments, the average time to establish the phase-lock condition is 15.5 time units. The time ranges from 11.6 to 19.6 depending on the distribution of sensor nodes, location of the core node, and phase of sensor nodes. The histogram is shown in Fig. 10. The time to reach the stable phase-lock condition can be reduced by using a set of larger $a$ and $b$ satisfying Eq. (3). For example, with $a = 0.05$ and $b = 0.6$, the minimum, average, and maximum
5.96, 8.10, and 10.7, respectively.

We confirmed that traveling waves can be formed in a wireless network with addition, movement, and removal of sensor nodes, and failed nodes, although figures are not shown for space limitation. For example, when new nodes are deployed at random locations after the phase-lock condition is established for information diffusion, they initially emit messages independently of their location. When a new sensor node receives a message from an existing sensor node, it sets the level value as the received value plus one. However, its timer has not been adjusted well yet. Therefore, the phase-lock condition of neighboring sensor nodes is lost as in the case of timer errors. However, as time passes, they begin to behave in synchrony with sensor nodes at the same distance from the core node and the phase-lock condition is re-established in the whole sensor network.

6. Conclusion and Future Work

In this paper, we first investigated initial conditions that lead to a desired form of traveling wave regardless of the initial phase of oscillators in a pulse-coupled oscillator model. Next, we proposed a fully-distributed and self-organizing communication mechanism in wireless sensor networks. Through simulation experiments, we confirmed that our scheme can gather or diffuse information in accordance with application’s requirements in a dynamic wireless sensor network.

As future research work, we plan to implement our mechanism using off-the-shelf sensor to verify the practicality of our mechanism. In an actual environment, radio signals are unstable and unreliable. Due to collisions among synchronized transmission of messages from sensor nodes at the same distance from a core node, i.e., with the same level value, messages will be lost or delayed. In [8], we implemented another PCO-based data gathering scheme where all sensor nodes of the same level value behave in synchrony and confirmed that it worked as expected with some additional mechanisms to solve the instability and unreliability of radio communication. We consider the mechanism proposed in this paper can also work well with a similar approach.

Acknowledgment

This work was partly supported by “The 21st Century Center of Excellence Program: New Information Technologies for Building a Networked Symbiosis Environment” and a Grant-in-Aid for Scientific Research (A)(2) 16200003 from The Japanese Ministry of Education, Culture, Sports, Science and Technology.

References


