Computer Communications 35 (2012) 661–669

Contents lists available at SciVerse ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Energy efficient self-organizing control for wireless sensor networks inspired by calling behavior of frogs

Akira Mutazono[a], Masashi Sugano[b,*], Masayuki Murata[a]

[a] *Graduate School of Information Science and Technology, Osaka University, Japan*
[b] *School of Comprehensive Rehabilitation, Osaka Prefecture University, Japan*

ARTICLE INFO

Article history:
Available online 28 September 2011

Keywords:
Wireless sensor network
Self-organized control
Antiphase synchronization
Bio-inspired control
Satellite behavior

ABSTRACT

One of the most challenging research tasks in the field of wireless sensor networks is controlling the power consumption of batteries and prolonging network lifetime. For sensor networks that consist of a large number of sensor nodes, research on bio-inspired self-organization methods has attracted attention due to the potential applicability of such methods. In this paper, we focus on the calling behavior of Japanese tree frogs. They are known to make calls alternately with their neighbors in order to raise the probability of mating. This behavior can be applied to phase control that realizes collision free transmission scheduling in wireless communication. These frogs also display a type of behavior known as satellite behavior, where a frog stops calling once it detects the calls of other neighboring frogs. This behavior can be applied in the design of an energy-efficient sleep control mechanism that provides adaptive operation periods. We propose a self-organizing scheduling scheme inspired by Japanese tree frog calling behavior for energy-efficient data transmission in wireless sensor networks. Simulation results show that our proposed sleep control method prolongs network lifetime by a factor of 6.7 as compared with the method without sleep control for a coverage ratio of 80%.

1. Introduction

Self-organized control inspired by biological systems has been receiving considerable attention as a promising concept for realizing robustness, scalability, and adaptability [1,2]. Biological systems always change and adapt in response to continuous changes in the environment. Each component of a biological system is governed by local interactions with neighbors, not controlled by a specific leader. Thus, the entire system can respond to changes in a coordinated manner in spite of the selfish behavior of individual components. Such simple mechanisms bring a cognitive function to the system as a whole, and self-organized control provides adaptability and robustness [3]. A number of methods have been proposed where the advantages of biological systems are applied to computer networks in fields such as routing [4] and clustering [5]. For example, in the field of time synchronization, pulse-coupled oscillators (PCO) [6] are known to mimic the behavior of fireflies, which flash in unison with their neighbors.

Although most existing research on the PCO model focuses on simultaneous synchronization, we hypothesize that antiphase synchronization of transmission scheduling can reduce packet loss caused by collisions on the wireless channel. As a possible mechanism for realizing antiphase synchronization, we focus on the calling behavior of Japanese tree frogs (*Hyla japonica*) [7,8], in particular *advertisement calling*. One of the main purposes of the calling behavior of this species of frog is for the male to attract females. If a male calls simultaneously with others, the female will have difficulty in distinguishing between callers, and therefore they shift the timing of their calls. This interesting aspect of frog calling behavior may be applicable to various fields [9]. Accordingly, we formulate the advertisement calling behavior by using the pulse-coupled oscillator model, and it is applied in phase control for antiphase synchronization, as well as in transmission scheduling for wireless communication with the aim of avoiding transmission failure. Conventional scheduling protocols have problems with the overhead for adjusting their schedule and lack adaptability since the schedule is fixed rather than responsive to environmental changes. However, self-organizing scheduling based on frog calling is expected to solve these problems.

In addition to properties such as robustness and scalability, energy conservation is critical in sensor networks since each node has access to a limited amount of energy provided by a battery and the battery may be difficult to change or charge depending on node deployment. To reduce energy consumption, sleep control is effective, and many studies on distributed sleep control techniques have been actively carried out [10,11]. We focus on an interesting behavior called *satellite behavior*, which is characteristic to

---

* Corresponding author.
E-mail address: sugano@rehab.osakafu-u.ac.jp (M. Sugano).

0140-3664/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2011.09.013

Japanese tree frogs. When there are many male frogs competing to mate with a female, some males stop calling based on a comparison between the merits of obtaining a higher probability of mating and the demerits of losing energy by calling. We apply this behavior to sleep control in wireless sensor networks. The decrease in the per-unit cost of sensor nodes is expected to allow for massive deployments in the future. In such situations, nodes will be deployed non-uniformly and redundantly meaning that some nodes do not need to perform active sensing continuously. By implementing control analogous to the satellite behavior, redundant nodes turn off the transceiver power and are thus able to conserve energy. In this scheme, each redundant node is subjected to sleep control according to the overlap of the sensing regions between neighboring nodes, and the number of active nodes can be adjusted without sacrificing coverage. In addition, efficient data gathering is achieved by adjusting the number of active nodes in consideration of the traffic load.

In this paper, we propose a self-organizing transmission scheduling scheme inspired by the calling behavior of Japanese tree frogs. First, we model the calling behavior of frogs mathematically by PCO, and describe the scheduling technique by antiphase synchronization. Next, we explain the proposed algorithm based on satellite behavior, and present evaluation results from simulation experiments.

The outline of this paper is as follows. Section 2 presents related work. Section 3 introduces details of the mechanisms of the phase control method based on the alternating calling behavior of frogs. Then in Section 4, we describe in detail the mechanism of the proposed sleep control method based on the satellite behavior of frogs. Section 5 presents the results of numerical simulations of single-hop networks. In Section 6, we state our conclusions and discuss possible extensions of this research.

2. Related work

2.1. Antiphase synchronization for scheduling

Many studies on time synchronization using the pulse-coupled oscillator model have been performed [12–14]. Those works aim at adjusting the phase of oscillators in unison; however, synchronization that shifts the phase of oscillators at certain intervals has not been previously considered in detail. We refer to conventional simultaneous synchronization as in-phase synchronization and call alternating synchronization of our target as antiphase synchronization. An example of in-phase synchronization is the phenomenon of simultaneous flashing of fireflies, and an example of antiphase synchronization is a string of Christmas lights on which bulbs of two colors flash alternately.

Antiphase synchronization is effective for sharing a resource. Round-robin scheduling assigns equal time slices to processes in a waiting state in order and without prioritization. This method is considered fair scheduling since the resource is allocated to all the processes equally. In the field of wireless communication, time division multiple access (TDMA) is another type of antiphase synchronization which divides the access period into fixed slots and assigns the frequency used for communication. In TDMA, since it is not necessary to check the status of a channel, delay is small and transmission speed is stable [15]. Furthermore, if antiphase synchronization is applied to a multi-hop network, collision in the MAC layer in the wireless sensor network can be avoided.

Some studies on antiphase synchronization have been conducted. DESYNC [16,17] is a distributed antiphase synchronization method proposed by Nagpal et al. The firing time is adjusted by each node considering its own last and next firing so that the phase offsets of firings become equal. Even when there are many nodes, iteration of interactions leads to the whole network being in an antiphase synchronized state. However, timing adjustment in this method relies on information from only two nodes, and this structure is not effective in a multi-hop network. Stankovic [18] proposed another antiphase synchronization method. This method adjusts the firing time for rare event detection considering the distribution of sensing region. However, this method requires considerable computational resources for formulating complex polynomial function and location information of the neighboring nodes is necessary for accurate antiphase synchronization. Phase diffusion time division (PDTD) [19] is a type of antiphase synchronization method that performs in a self-organizing manner. This method solves the hidden terminal problem by performing antiphase synchronization between nodes within an interaction range that is twice as large as the communication range.

2.2. Sleep scheduling for energy conservation

The main purpose of sleep control is to switch off redundant nodes in order to reduce energy consumption. As a result, nodes can operate for longer periods of time and the network lifetime is extended. Energy efficiency is especially critical in wireless sensor networks. Normally, a sensor node is powered by a battery of limited capacity, and it may be difficult to charge or replace batteries due to the deployment of the nodes. Moreover, interference occurs if many nodes communicate simultaneously in a dense network, and sleep control can solve this problem by limiting the number of active nodes. Sleep control is also effective for minimizing delays by adjusting the transmission schedule when performing data collection.

For the above reasons, sleep control has been the subject of extensive research. Packet collision, the reception of unrelated packets, the overhead of control packets, and redundant idling time are often regarded as causes of energy waste, and a sleep mechanism is proposed to reduce such problems. S-MAC [20] is the most well-known sleep control method. In that method, packet collision is avoided by synchronizing the active time for reducing the idling time, and a virtual carrier senses by means of request-to-send (RTS) and clear-to-send (CTS) packets. PEAS [21] is another sleep method, in which nodes broadcast probe packets regularly and enter a sleep state whenever the number of active nodes in the neighborhood exceeds a certain threshold. The delay is reduced by adaptively setting the sleep time according to the number of active nodes. A sleep control method based on the remaining energy has also been proposed [22]. This technique becomes effective when nodes have a wide distribution of remaining energy. van Greunen et al. [23] proposed a sleep control method for minimizing the queuing delay in data transfer. In addition, the DMAC protocol [24] first builds a network tree and nodes transmit data following shifted timing based on the hop count from the sink, thus reducing the delay between event detection and the transmission of detected data to the sink. There are also coverage-based sleep control methods. For example, CCP [25] guarantees various degrees of coverage depending on the application and reduces the energy consumption while satisfying the requirements of high coverage and connectivity. Thus, the function of sleep control changes according to the requirements. In this paper, we propose sleep control based on the satellite behavior of Japanese tree frogs for the purpose of energy conservation in redundant nodes and the prevention of packet collision while maintaining high coverage.

3. Antiphase synchronization control

The outline of phase control is shown in Fig. 1. The frog calls by making a sound for a certain period of time and then becomes

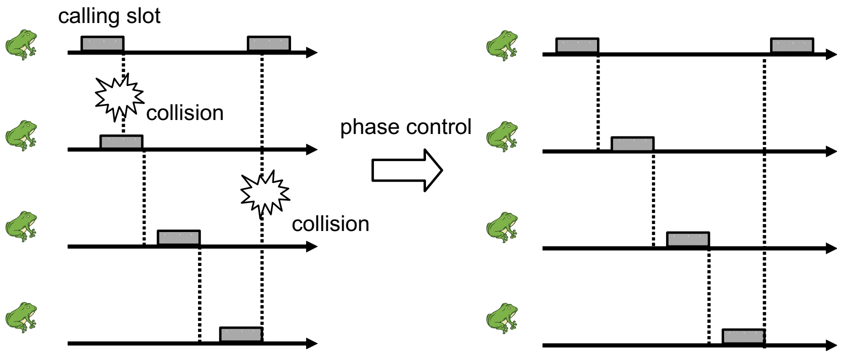


Fig. 1. Outline of phase control which reduces the occurrence of transmission failure by adjusting the transmission timing.

quiet before repeating the call. If two or more male frogs call at random, their calls might overlap. In such a case, the calls interfere with each other and the female frog (the mating partner) cannot distinguish between the callers. Therefore, each male frog shifts the timing of its calls by listening to the calls of other frogs so as to avoid such overlap. After all frogs establish this interaction pattern, call alternation without interference is achieved within the group.

Pulse-coupled oscillators are used as models of various synchronization mechanisms in biology. Here, we formulate frog calling behavior with pulse-coupled oscillators. Each oscillator with a firing frequency ω has a phase $\phi \in [0, 2\pi]$ that changes with time. When the phase reaches 2π, the oscillator fires and returns the phase to the initial value ($\phi = 0$). Oscillator j which is coupled with firing oscillator i receives a stimulus and changes the firing frequency of the next turn in accordance with the phase offset $\Delta_{ji} \in [0, 2\pi]$ between the coupled oscillators. The oscillator does not change the firing frequency immediately after receiving the stimulus; instead, it stores the size of the stimulus and changes the firing frequency after firing its own stimulus.

$$\omega_j = \frac{d\phi_j}{dt} \tag{1}$$

$$\Delta_{ji} = \phi_j - \phi_i \tag{2}$$

$$\omega_j^+ = \omega_j + g(\Delta_{ji}) \tag{3}$$

where $g()$ is the phase shift function for generating the repulsive force that shifts the phase away from that of other oscillators. Aihara et al. [26] proposed the following phase shift function:

$$g(\Delta) = \alpha \sin \Delta \tag{4}$$

where the coupling coefficient of a pulse-coupled oscillator model is $\alpha > 0$. When $\Delta_{ji} < \pi$, then $g(\Delta_{ji}) > 0$ and oscillator j increases the firing frequency to extend the phase offset with respect to oscillator i. On the contrary, when $\Delta_{ji} > \pi$, then $g(\Delta_{ji}) < 0$ and oscillator j decreases the firing frequency in order to spread the phase offset with respect to oscillator i. After these interactions, the oscillators are assumed to be in a stable antiphase synchronized state when the following conditions are fulfilled (Fig. 2):

$$\Delta_{ij} = \Delta_{ji} \tag{5}$$

$$g(\Delta_{ij}) = g(\Delta_{ji}) = 0 \tag{6}$$

We then consider the group N, in which n oscillators are coupled with each other. When oscillator j fires at time t_j ($t_1 < t_2 < \cdots < t_n$), it changes the firing frequency ω_j as follows:

$$\Delta_{ji} = \phi_j(t_i) - \phi_j(t_i) \tag{7}$$

$$\omega_j^+ = \omega_0 + \sum_{k \in N} g(\Delta_{jk}) \tag{8}$$

When the phase offsets between oscillators that fire consistently are all equal and the repulsive force of all oscillators is negated, the group is assumed to be in a stable antiphase synchronized state. These conditions are described below together with the case of two oscillators

$$\Delta_{12} = \Delta_{23} = \cdots = \Delta_{n1} \tag{9}$$

$$\sum_{k \in N} g(\Delta_{1k}) = \sum_{k \in N} g(\Delta_{2k}) = \cdots = \sum_{k \in N} g(\Delta_{nk}) \tag{10}$$

Two or three oscillators can be antiphase synchronized with phase shift function Eq. (4) (Fig. 3(a) and (b)). However, this function cannot achieve antiphase synchronization for four or more oscillators since they are divided into groups of two oscillators and groups of three oscillators (Fig. 3(c) and (d)). This is caused by the phase shift function, which is a symmetric function, and the repulsive force is negated in situations where condition Eq. (9) is not satisfied, despite condition Eq. (10) being satisfied and the oscillators converging to a stable state. The stimulus should be weighted depending on the phase distance δ in order to resolve this problem. The smaller the phase distance δ between the coupled oscillators, the stronger the oscillators should be in order to receive the stimulus. For this reason, we adopt the following equations:

$$\delta(\Delta) = \min\{\Delta, 2\pi - \Delta\} \tag{11}$$

$$g(\Delta) = \alpha \sin(\Delta) \exp(-\delta(\Delta)) \tag{12}$$

By using this phase shift function Eq. (12), conditions (9) and (10) are always satisfied, regardless of the number of oscillators. Fig. 4 shows the process of antiphase synchronization between 10 oscillators. The oscillators' phases, which are discrete in the initial state, are shifted to an antiphase synchronized state through the interactions between coupled oscillators. The phase offset between consecutive oscillators becomes approximately the same at a time of 1.0 s. After that point, the oscillator receives positive and negative stimuli that cancel each other out, and the group maintains a stable state.

We next evaluate the coupling coefficient α, which is an important parameter of the pulse-coupled oscillator model. To obtain suitable parameter settings in accordance with the number of nodes, we estimate the average error after a certain period (20 s). This value shows the average of the phase offset between nodes; the smaller the average error, the higher is the synchronization accuracy. Fig. 5(a) shows that the average error of 10^{-2} becomes the synchronization threshold value, below which the accuracy

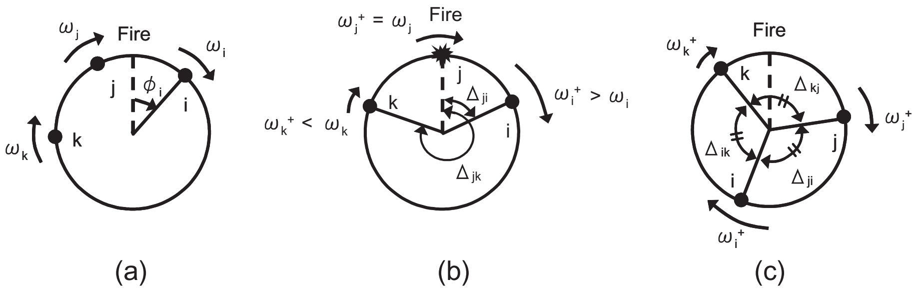


Fig. 2. Phase control mechanism. (a) Each oscillator has its own phase and firing frequency. (b) Oscillator *i* receives positive stimulus and increasing the firing frequency, oscillator *k* receives negative stimulus and decreases the firing frequency. (c) After a number of iterations, the phase offset between each oscillator becomes equal and antiphase synchronization is realized.

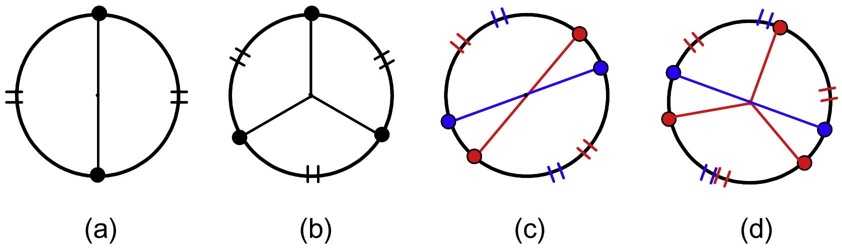


Fig. 3. Difficulty in antiphase synchronization. Four or more oscillators are divided into groups of two oscillators and groups of three oscillators. Antiphase synchronization is realized within each group, but not in the system as a whole.

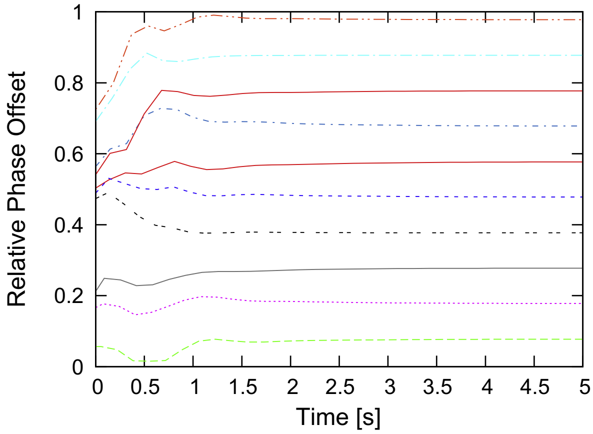
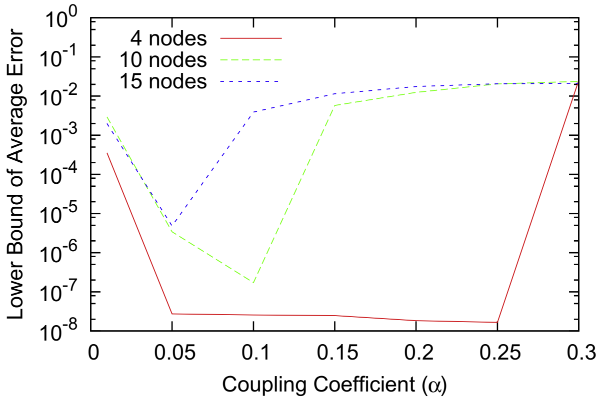
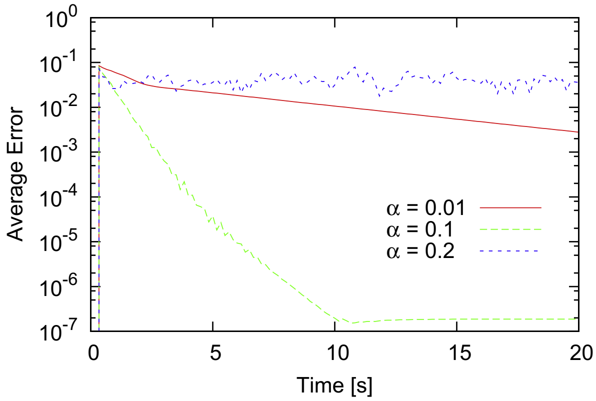


Fig. 4. Transition of relative phase offset.

of synchronization becomes higher with time and above which the phase continues to fluctuate and does not converge to a stable state. Additionally, the wide range of the coupling coefficient values enables the network to reach a stable state when a small number of nodes are deployed, and it becomes difficult to converge to the stable state if the value of the coupling coefficient is too large. This result is a consequence of the number of coupled nodes; in other words, the stimulus becomes stronger as a node becomes coupled with more nodes and the coupling coefficient becomes larger. Hence, the simulation shows that over-stimulation disturbs convergence to a stable state. On the contrary, although small values of the coupling coefficient require longer synchronization times, the network steadily approaches a stable state (Fig. 5(b)). These results indicate that antiphase synchronization requires the coupling coefficient to be set adaptively. The choice of coupling coefficient also depends on the requirements of the particular application; for example, delay-tolerant applications should have



(a) Relation between coupling coefficient α and lower bound of average error.



(b) Transition of average error with 10 nodes.

Fig. 5. Setting of coupling coefficient α.

a small coupling coefficient and accuracy-tolerant applications should have a large coupling coefficient. The number of nodes and the data transmission interval also affect the setting of the coupling coefficient. Various factors must be considered, and those factors are assumed to change constantly. Therefore, setting a static coupling coefficient is not sufficient, and it is required that the parameter be set dynamically for each node in accordance with the number of nodes and the amount of traffic in a self-organizing manner. However, this problem is beyond the scope of the present work and thus left for future study.

4. Sleep control based on frog satellite behavior

Next, we explain our proposed sleep control method, which is inspired by frog satellite behavior. Where the details of the satellite behavior were discussed, we consider the following three factors:

- Territory.
- Number of competing frogs in the area.
- Body size.

Considering these three factors, the frog decides whether to adopt satellite behavior. We explain each condition for the decision in Fig. 6. The red circles represent the target frogs that are considering their behavior, and the dotted lines indicate their respective territories. The frogs are located in a paddy field, and each frog is assumed to be able to hear all other frogs calling.

1. Three frogs (denoted A, B, and C) are already calling, and two other frogs are evaluating the environment. Frog D on the left first checks that in its territory there is no calling frog, next checks that the number of calling frogs in the paddy field is not exceedingly large, and then begins to produce calls. On the other hand, frog E on the right adopts satellite behavior as it finds two other frogs (A and B) calling in its territory.
2. A new large frog F appears and evaluates its surroundings. It finds frog C calling in its territory. However, after comparing the size of their bodies, it concludes that the probability of winning the competition is large and begins to call. We refer to this behavior as *interception*.
3. The small frog C detects the presence of a stronger frog in its territory and adopts satellite behavior in order to avoid competing.

Applying this mechanism to sensor networks, the territory of a frog corresponds to the sensing region: the number of competing frogs in the area, the number of access nodes; and the body size, the remaining energy. The following targets are achieved with such a sleep control mechanism.

- Energy conservation for redundant nodes.
- Distribution of coverage.
- Assignment of bandwidth (i.e., access period) for reducing packet collisions.

In sensor networks, a large number of nodes are assumed to be deployed randomly, in which case several nodes might be arranged closely together within each other's range. It is not necessary for all such neighboring nodes to perform sensing in the same region since a single node is sufficient in that case. Energy consumption is reduced when redundant nodes adopt satellite behavior and enter sleep mode. Additionally, the proposed method, in which nodes enter sleep mode not randomly but rather according to the state of neighboring nodes, is energy efficient while satisfying the requirement of high network coverage. Although antiphase synchronization is performed accurately, there is a limit to the number of nodes that can transmit in turns depending on the frequency of data transmission and the data packet size. By performing sleep control and considering these factors, packet collision caused by excessive traffic can be prevented.

Each sensor node maintains two states, *active* and *satellite*. A node checks the state of the network regularly, and the node changes its state accordingly. In the active state, the node periodically executes the network check process immediately before data transmission, whereas in the satellite state the node checks the state of the network after a sleep period T_s. We call the node that executes the network check a *checking node* (CN); the sensing region, a *territory*; and another node in the territory, a *territory competing node* (TCN). It is possible for a node to take the place of another node that has little remaining energy even if many nodes are already in the active state. We refer to the node with the lowest remaining energy in the active state as a *bandwidth competing node* (BCN). If the remaining energy of CN is larger than the remaining energy of BCN, CN becomes active instead of BCN. The flow of the check process is shown in Fig. 7:

1. CN checks in its sensing region whether there is a TCN.
2. If CN finds a TCN, it compares the remaining energy with that of TCN in order to evaluate the probability of interception. If the remaining energy of CN is higher, CN determines that it will not lose the competition and enters the active state for transmission; otherwise it enters the satellite state in order to save energy.
3. If there is no TCN, CN evaluates the available bandwidth. If there is sufficient bandwidth, CN determines that transmission is possible and enters the active state.
4. If the bandwidth is being used by active nodes, the CN compares its remaining energy with PCN for interception. If CN has higher remaining energy, it enters the active state; otherwise it enters the satellite state.

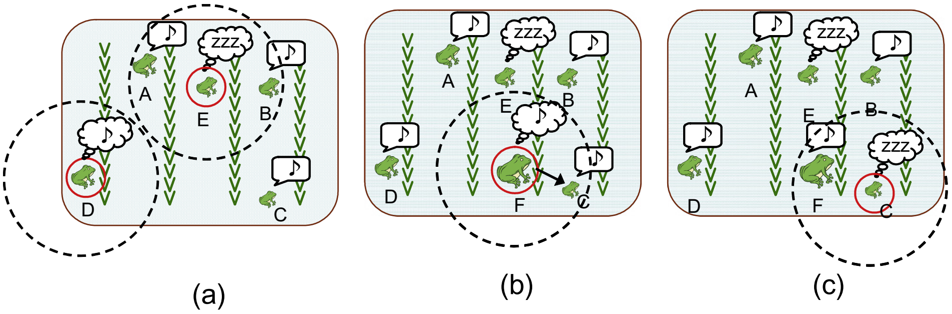


Fig. 6. Outline of sleep control. Red circles denote the target frogs and dotted lines indicate their respective territories. Each frog in the figure is drawn to scale to show body size. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

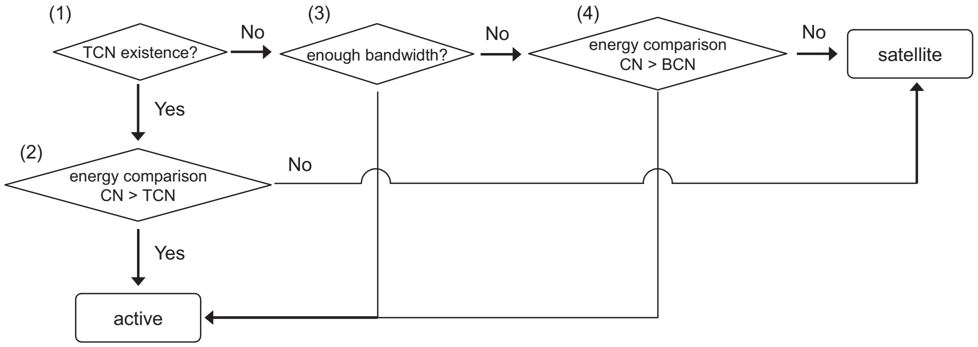


Fig. 7. Flow chart of network check process.

Active nodes periodically perform data transmission and satellite nodes sleep to conserve energy. In order for a satellite node to enter the active state, its remaining energy must be higher than that of PCN. In addition, the higher the remaining energy of the node, the more often it performs the network check process in order to increase the probability of entering the active state. For these reasons, a satellite node sets the sleep period T_s based on a standard sleep time T_{std} by considering its own remaining energy E_{CN} and the remaining energy E_{BCN} of BCN

$$T_s = T_{std} \cdot \exp\left(1 - \frac{E_{CN}}{E_{BCN}}\right) \tag{13}$$

In this manner, the proposed mechanism implements sleep control, which conserves energy by ensuring that redundant nodes enter sleep state, reduces collisions by adaptive assignment of access periods, and prolongs network lifetime by performing sleep control considering the remaining energy of nodes.

5. Evaluation

5.1. Simulation setup

In these simulations, sensor nodes are deployed randomly in a monitoring region with a radius of 10 m, and the nodes set the data transmission timing on the basis of the phase, which is assigned randomly in the initial state. The communication range of a node is assumed to be 20 m, and the nodes can communicate with all other nodes in the network. The node carries out sensing of a monitoring region with a radius of 5 m every 0.16 s and transmits the sensing data to the base station, referred to as a sink, at a transmission speed of 50 kb/s. CSMA/CA is used for the transmission protocol of the MAC layer. A data packet includes sensing information and a timestamp that indicates the delay caused by the backoff of CSMA/CA, and the packet size is set to 400 bits. Therefore, 8 ms is required in order to transmit one data packet, and the transmission node takes exclusive control over the communication band during that period. The standard sleep time T_{std} in the sleep control scheme is set to 3.2 s. We use the following evaluation metrics:

- Transmission failure probability.
  Probability of transmission failure induced by timeout due to backoff in CSMA/CA during a communication attempt of a node.
- Data collection ratio.
  Ratio of the amount of data reaching the sink to all data sent to the sink from a node.
- Coverage ratio.
  Ratio of the region with 1-coverage to the entire monitoring region.
- Energy utilization ratio.
  Ratio of the mean remaining energy of all nodes to the mean remaining energy of all active nodes. When this value is close to 1, it is considered that the task is divided fairly and the energy consumption is distributed evenly between nodes.

5.2. Performance of the proposed sleep control mechanism

The objectives of the proposed sleep control mechanism are (1) energy conservation in redundant nodes, (2) even distribution of coverage, and (3) assignment of access periods to reduce packet collisions. Perfect antiphase synchronization cannot ensure alternating data transmission for all nodes when the number of nodes is large and the access period is narrow. In this case, if there are many active nodes, the overlap of access periods is prevented by making redundant nodes enter sleep mode, and data transmission becomes free from transmission failure. In other words, sleep control brings adaptability to phase control for reliable transmission. After this section, we describe the results for 20 nodes that were deployed in the monitoring region in order to evaluate the efficiency of the proposed sleep control mechanism.

As one of the principles of sleep control, a node with high remaining energy attempts to replace a node with low remaining energy and subsequently performs sensing. This principle ensures a fair distribution of energy consumption between nodes. Fig. 8 shows the number of active nodes and the energy utilization ratio. At the beginning, seven nodes become active for sensing and other nodes enter sleep mode since it is impossible for all 20 nodes to transmit during the same transmission period. The value of the energy utilization ratio decreases as the seven active nodes consume energy. Then, after 50 s, interception is performed by the satellite nodes, the nodes change state, and the energy utilization ratio rises

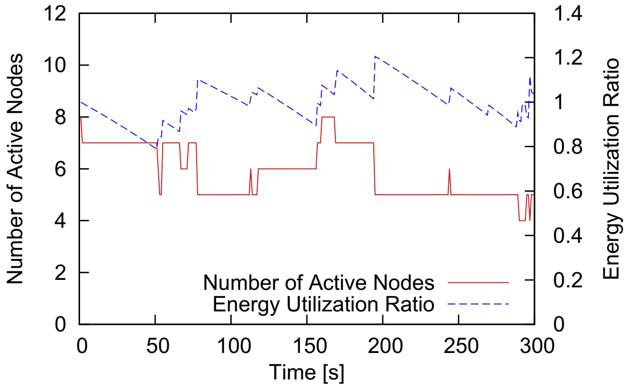
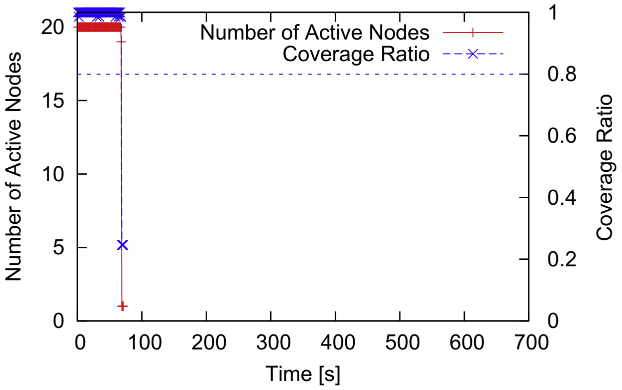
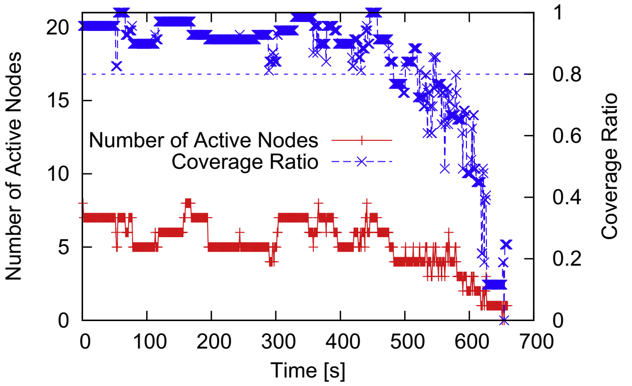


Fig. 8. Number of active nodes with respect to the energy utilization ratio.

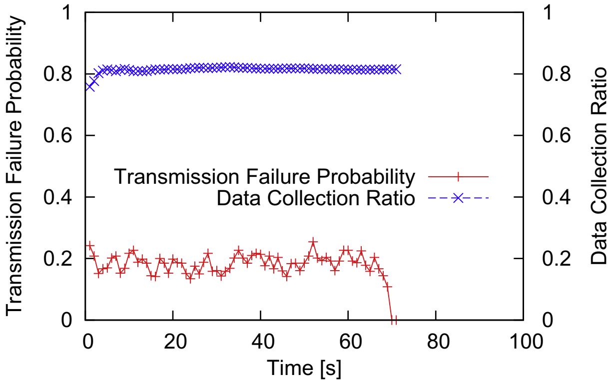


(a) Case where phase control is implemented without sleep control.

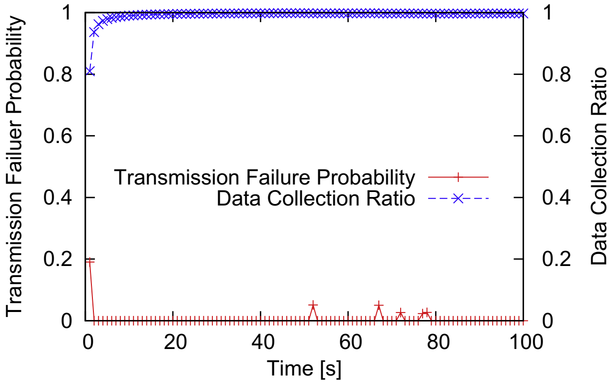


(b) Case where both phase control and sleep control are implemented.

Fig. 9. Comparison of coverage ratio.



(a) Case where phase control is implemented without sleep control.



(b) Case where both phase control and sleep control are implemented.

Fig. 10. Comparison of data collection ratio.

due to the participation of fresh nodes. After another 50 s, the nodes keep changing their state depending on the remaining energy, and thus the value of the energy utilization ratio is maintained at 1.0.

We observe the types of changes caused by sleep control through a comparison of the phase control method with and without sleep control. Fig. 9 shows the results for the number of nodes and the coverage ratio as a function of time. The method with sleep control provides a large sensing region over a long period of time with a small number of active nodes. On the contrary, the method without sleep control can achieve a high coverage ratio because sensing is performed by all nodes; however, the energy of the nodes is depleted within a short period of time. The sleep control mechanism prolongs the network lifetime by a factor of about 6.7 for a coverage ratio of 80%.

We evaluate the data collection ratio for the two methods in Fig. 10. In the method without sleep control, data transmission failure occurs without the assignment of access periods to the nodes. As a result, the sink cannot collect all data. On the other hand, the method with sleep control assigns access periods only to the active nodes and realizes reliable data transmission. Although transmission failure occurs after the cycle of alternation of nodes is completed, the number of failures is sufficiently small that the data collection ratio is unaffected. The combination of phase control and sleep control reduces the occurrence of transmission failure by adaptive assignment of access periods, prolongs the network lifetime, and satisfies the requirement of high coverage.

Next, we examine the relation between the number of sensor nodes and the network lifetime through simulation experiments. We define the period from the start of a simulation until the sensor coverage ratio becomes less than 80% as the "network lifetime". In other words, we assume that the network ceases to be operational when the sensors' batteries are completely depleted or when the coverage ratio is less than 80%. We set the initial energy stored in the batteries of all the sensors to 6 J. Fig. 11 shows the average value and the 99% confidence interval of network lifetime, as obtained through the simulation. Under these simulation conditions, the number of sensor nodes required to achieve a coverage ratio of 80% is about 15 nodes. Therefore, if 15 or more redundant sensors are installed, as long as each sensor suitably switches between operation and sleep according to the remaining energy, the whole network will operate by virtue of the satellite function in our proposed technique. In this simulation, we install up to 45 nodes, about 3 times the required numbers of nodes, and investigate network lifetime. The network lifetime increases as the number of nodes is increased, but the increase is not linear and the slope decreases as the number of nodes becomes large. Since we install sensor nodes at random in these simulations, the allocation density of sensors is not uniform. Therefore, the energy of a sensor in a region with low allocation density and few adjacent sensors will be consumed most rapidly. This is one reason why network lifetime does not increase linearly with the number of sensors. Moreover, since the collisions of packet will also increase as the allocation density of sensors becomes high, the resulting power consumption will increase, which is also a factor in the nonlinear increase in network lifetime. In order to determine the number of sensors required for a certain network lifetime, this relation must be taken into consideration.

5.3. Evaluation of the robustness of the proposed technique

In a sensor network, since many nodes are installed in different environments, node properties during operation are not homogeneous. Moreover, degradation of the communication quality by

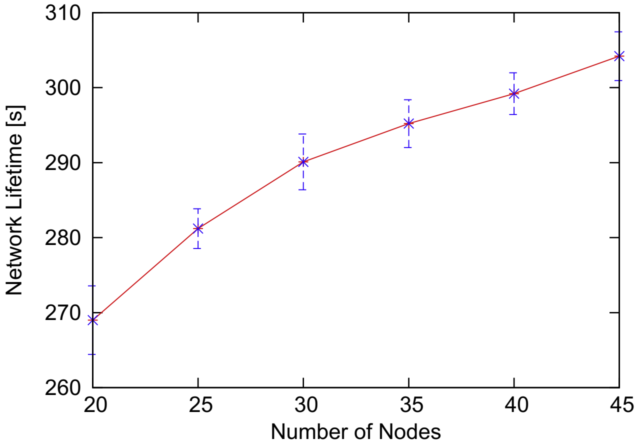


Fig. 11. Relation between number of sensor nodes and network lifetime.

noise or interference also has a large impact on network performance. Accordingly, through simulations, we investigate the impact of these factors on network lifetime when the proposed technique is used in network operation.

First, we evaluate the effects of the proposed technique in the case that the initial energies of the batteries were different. Fig. 12 shows the average network lifetime, as well as the 99% confidence interval, for different variations in the initial energy from 0% to ±50%. The network lifetime decreases as the variation in initial power becomes large, but the decrease is almost linear. Moreover, in this simulation, even though the nodes are positioned at random, the variation in the network lifetime is small for every simulation trial. Therefore, variance of the operational period of each node, according to remaining energy, is successfully accomplished with the proposed technique. Battery performance differs greatly depending, for instance, on the temperature of the environment where sensor nodes are installed. This property of the proposed technique should prove useful when sensors on a network must be installed in various environments with different conditions.

Next, we investigate the effects of changes in the packet loss rate when the proposed technique is used. Fig. 13 shows the network lifetime for packet loss rates from 0% to 50%. As shown in this figure, the network lifetime decreases as the packet loss rate increases. However, even for a packet loss rate of 50%, network lifetime decreases only about 20%. This result is because the change in remaining energy is gradual compared with the frequency of packet transmission and reception, and because sleep control is not strongly affected by packet loss.

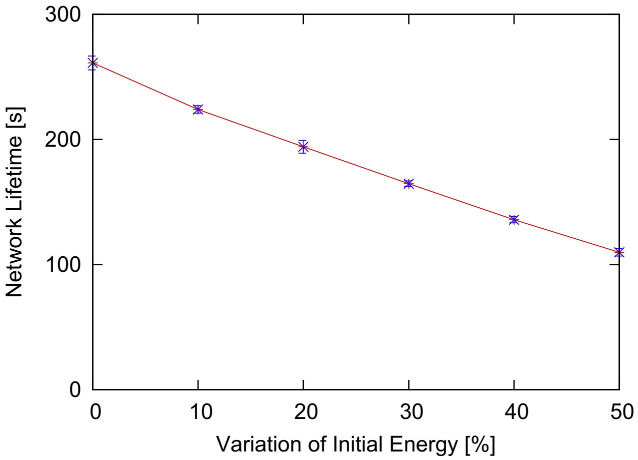


Fig. 12. Effect of variation in the initial energy of batteries.

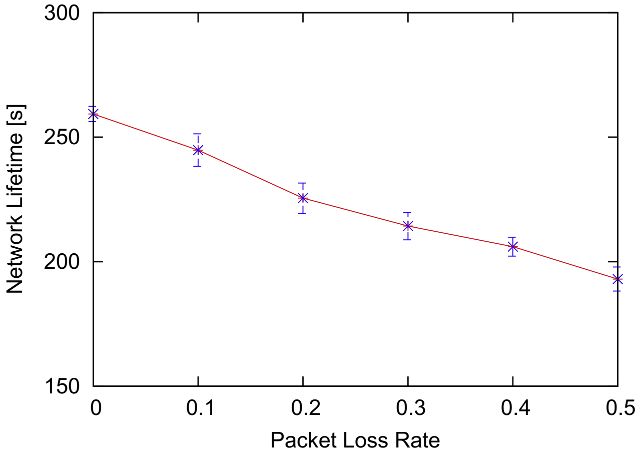


Fig. 13. Effect of packet loss rate on network lifetime.

6. Conclusion and possible extensions

In this paper, we introduced a self-organizing scheduling scheme inspired by the calling behavior of Japanese tree frogs as a method for reducing energy consumption as well as packet collision while satisfying the requirement of high coverage. We performed an evaluation through computer simulations of a single-hop network for the proposed functions by implementing a sleep control method inspired by the frogs' satellite behavior. The simulation results indicated that the sleep control enables adaptive assignment of access periods, fair distribution of coverage, and extension of the network lifetime.

Although this paper presents the application of the proposed scheduling scheme to a single-hop network, the possibility for expanding this scheme to the multi-hop case and applying it to larger networks should be explored. For that purpose, we aim at establishing a mathematical model of satellite behavior. From the complex structure of the satellite behavior, we applied only the operation of stopping calling in accordance with the strength of the individuals present in the surroundings in sleep control of sensor networks. However, actual frogs intercept females attracted to stronger males or move to another location where stronger individuals are not present. We consider that such behavior will also be applicable to wireless sensor networks in various situations.

Acknowledgment

This research was supported in part by the Sekisui Chemical Grant Program for Research on Manufacturing Based on Learning from Nature and The Okawa Foundation.

References

[1] H. Kitano, Biological robustness, Nature Reviews Genetics 5 (11) (2004) 826–837.
[2] M. Meisel, V. Pappas, L. Zhang, A taxonomy of biologically inspired research in computer networking, Computer Networks 54 (2010) 901–916.
[3] F. Dressler, Self-organization in Sensor and Actor Networks, Wiley, 2007.
[4] Y. Zhang, L. Kuhn, M. Fromherz, Improvements on ant routing for sensor networks, in: Proceedings of the 4th International Workshop on Ant Colony Optimization and Swarm Intelligence (ANTS 2004), 2004, pp. 154–165.
[5] H. Chan, A. Perrig, ACE: an emergent algorithm for highly uniform cluster formation, in: Proceedings of the 1st European Workshop on Wireless Sensor Networks (EWSN 2004), 2004, pp. 154–171.

[6] R.E. Mirollo, S.H. Strogatz, Synchronization of pulse-coupled biological oscillators, Journal on Applied Mathematics 50 (6) (1990) 1645–1662.

[7] I. Aihara, S. Horai, H. Kitahata, K. Aihara, K. Yoshikawa, Dynamical calling behaviors experimentally observed in Japanese tree frogs (*Hyla japonica*), IEICE Transactions on Fundamentals E90-A (2007) 2154–2161.

[8] I. Aihara, Modeling synchronized calling behavior of Japanese tree frogs, Physical Review E 80 (1) (2009) 011918.

[9] H. Hernandez, C. Blum, Distributed graph coloring: an approach based on the calling behavior of Japanese tree frogs. arXiv.org, <http://arxiv.org/abs/1011.5349v1 [cs.AI]>.

[10] A. Giusti, A.L. Murphy, G.P. Picco, Decentralized scattering of wake-up times in wireless sensor networks, in: Proceedings of the 4th European Conference on Wireless Sensor Networks (EWSN 2007), 2007.

[11] J. Niu, Z. Deng, Distributed self-learning scheduling approach for wireless sensor network, Ad Hoc Networks, in press. doi:10.1016/j.adhoc.2010.11.004.

[12] M.B.H. Rhouma, H. Frigui, Self-organization of pulse-coupled oscillators with application to clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (2) (2001) 180–195.

[13] G. Werner-Allen, G. Tewari, A. Patel, M. Welsh, R. Nagpal, Firefly-inspired sensor network synchronicity with realistic radio effects, in: Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems (SenSys 2005), 2005, pp. 142–153.

[14] A. Tyrrell, G. Auer, C. Bettstetter, A synchronization metric for meshed networks of pulse-coupled oscillators, in: Proceedings of the 3rd International Conference on Bio-Inspired Models of Network, Information and Computing Systems (BIONETICS 2008), 2008.

[15] J. Ma, W. Lou, Y. Wu, X.-Y. Li, G. Chen, Energy efficient TDMA sleep scheduling in wireless sensor networks, in: Proceedings of IEEE INFOCOM 2009, 2009, pp. 630–638.

[16] J. Degesys, I. Rose, A. Patel, R. Nagpal, DESYNC: self-organizing desynchronization and TDMA on wireless sensor networks, in: Proceedings of the 6th International Conference on Information processing in Sensor Networks (IPSN 2007), 2007, pp. 11–20.

[17] J. Degesys, R. Nagpal, Towards desynchronization of multi-hop topologies, in: Proceedings of the Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO 2008), 2008, pp. 129–138.

[18] Q. Cao, T. Abdelzaher, T. He, J. Stankovic, Towards optimal sleep scheduling in sensor networks for rare-event detection, in: Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN 2005), 2005.

[19] K. Sekiyama, Y. Kubo, S. Fukunaga, M. Date, Phase diffusion time division method for wireless communication network, in: Industrial Electronics Society, 2004, IECON 2004, 30th Annual Conference of IEEE, vol. 3, 2004.

[20] W. Ye, J. Heidemann, D. Estrin, Medium access control with coordinated adaptive sleeping for wireless sensor networks, IEEE/ACM Transactions on Networking 12 (3) (2004) 493–506.

[21] F. Ye, G. Zhong, J. Cheng, S. Lu, L. Zhang, PEAS: a robust energy conserving protocol for long-lived sensor networks, in: Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS 2003), 2003, pp. 28–37.

[22] M. Pearlman, J. Deng, B. Liang, Z. Haas, Elective participation in ad hoc networks based on energy consumption, in: Proceedings of the Global Telecommunications Conference (GLOBECOM 2002), vol. 1, 2002.

[23] J. van Greunen, D. Petrovic, A. Bonivento, J. Rabaey, K. Ramchandran, A. Vincentelli, Adaptive sleep discipline for energy conservation and robustness in dense sensor networks, in: Proceedings of the IEEE International Conference on Communications (ICC 2004), vol. 6, 2004, pp. 3657–3662.

[24] G. Lu, B. Krishnamachari, C. Raghavendra, An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks, in: Proceedings of the 18th International Parallel and Distributed Processing Symposium (IPDPS 2004), 2004.

[25] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, C. Gill, Integrated coverage and connectivity configuration in wireless sensor networks, in: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys 2003), 2003, pp. 28–39.

[26] I. Aihara, H. Kitahata, K. Yoshikawa, K. Aihara, Mathematical modeling of frogs' calling behavior and its possible application to artificial life and robotics, Artificial Life and Robotics 12 (1) (2008) 29–32.