Performance Evaluation of Intermittent Receiver-Driven Data Transmission on Wireless Sensor Networks

Daichi Kominami #1, Masashi Sugano *2, Masayuki Murata #3
Takaaki Hatauchi $4, Yoshikazu Fukuyama $5

Graduate School of Information Science and Technology, Osaka University
1-5, Yamadaoka, Suita-shi, Osaka 565-0871, Japan
{1 d-kominami, 3 murata}@ist.osaka-u.ac.jp

* *School of Comprehensive Rehabilitation, Osaka Prefecture University*
3-7-30, Habikino, Habikino-shi, Osaka 583-8555, Japan
2 sugano@rehab.osakafu-u.ac.jp

$ *Fuji Electric Systems*
1, Fujicho, Hino-shi, Tokyo 191-8502, Japan
{4 hatauchi-takaaki, 5 fukuyama-yoshikazu}@fujielectric.co.jp

***Abstract*—An intermittent transmission control in the MAC layer protocol is very important for managing the energy consumption of wireless sensor networks. This research focuses on an intermittent receiver-driven data transmission (IRDT) protocol in which communication starts when multiple receiver nodes transmit their own IDs intermittently and a sender node receives them. This method aims at prolonging network life time under the conditions that data-generating frequency is comparatively small. In this paper, we clarify the performance characteristics of this method by comparing it with the low power listing (LPL) method, which is a sender-driven protocol. By simulation, we show that IRDT can result in a higher reduction of energy consumption than LPL, especially at small loads. We also propose an improved IRDT scheme. While all nodes have equal and constant intermittent intervals in the original IRDT, the improved IRDT changes each node's intermittent intervals adaptively. This achieves more than 98% packet collection ratio and 50% lower power consumption than adaptive LPL, which also sets the intermittent interval adaptively.**

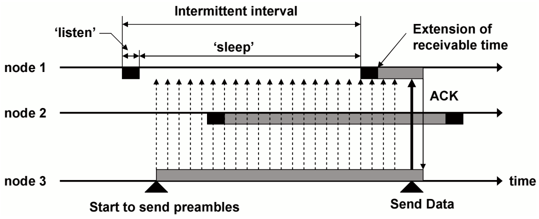
I. INTRODUCTION

Some important technical problems still need to be solved in wireless sensor networks, one of which is saving energy in sensor nodes with limited battery life. There are various approaches for saving energy, for example, miniaturizing sensor nodes, MAC with sleep control, multi-hop routing, etc.[1-4]. In this paper, we use one of the MAC layer approach i.e., intermittent operation. Because sleeping nodes consume significantly less energy than idling nodes, sensor nodes should enter the sleep state when they are not sending or receiving data so as to reduce energy consumption. However, in such case, nodes must control the time they wake up in order to communicate with each other.

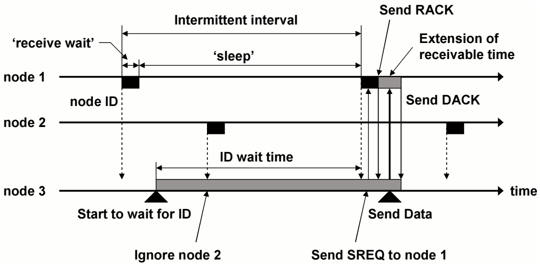
Control methods for intermittent operation are classified into two types: synchronous [3] and asynchronous [5, 6] type. A synchronous method uses a beacon to maintain synchronization of intermittent operations. The advantage of synchronization is there is less delay between a sensor node waking up and sending data, and the disadvantage is that sending a beacon regularly consumes power and causes interference. An asynchronous method is either sender-driven or receiver-driven, depending on which node initiates communication, the data sender or receiver. Regardless, nodes can start communication at any given point in time, but packet collisions must be controlled. Furthermore, in an asynchronous intermittent operation, the sender node waits in an idle state until the receiver node awakens. For these reasons, in terms of reducing energy consumption, the synchronous method is advantageous in a periodic data collecting system with comparatively high data-generating frequency. Meanwhile, the asynchronous type is superior in a system where packets are less frequently generated according to arbitrary timings. We target an application that needs high packet collection ratio and operates for about several years without the necessity of replacing the battery in a situation where data-generating frequency is comparatively small. Hence, in a case where sensor nodes with intermittent operation construct an ad-hoc network, the asynchronous method suits our application best.

The low power listening (LPL) protocol is a sender-driven asynchronous type of ad-hoc network system [6]. The intermittent operation of LPL is shown in Fig. 1(a). Receivable nodes 1 and 2 intermittently enter the 'listen' state and check the channel condition. If the channel is idle, they return to the 'sleep' state, and if busy, they change their own 'listen' state to the 'receive wait' state. After receiving data packets intended for them, they return acknowledge packets. When node 3 wants to send data to node 1, first, in order to make the channel busy, node 3 continuously sends preamble packets for a longer time than the intermittent interval (Fig. 1(a)). Next, after sending preamble packets, node 3 sends a data packet. There are many restrictions in LPL, i.e., each sender node occupies the channel during the fixed time by transmitting preamble packets, and each sender node has only a specific node with which communication is possible.

To lift the restrictions of LPL, we proposed the intermittent receiver-driven data transmission (IRDT) protocol in our previous work [7]. This protocol is for an actual product under development. By implementing IRDT, we are developing a meter reading system which can be operated with a battery for



(a) Low power listening (LPL)



(b) Intermittent receiver-driven data transmission (IRDT)

Fig. 1. Asynchronous intermittent transmission methods

a long period of time. In IRDT, a receiver-driven asynchronous system, receivable nodes send their own IDs to inform other nodes that they are ready to receive packets (Fig. 1(b)). A sender node waits for receiver IDs and when it gets an ID from an appropriate receiver, it establishes a link with the receiver and sends a data packet. Unlike in LPL, there is no occupancy of the channel in IRDT. Moreover, a sender node can select a receiver from one or more communication candidates, which improves reliability and reduces the active time sender side waiting for an appropriate receiver to wake up.

The performance characteristics of IRDT have not been clearly clarified yet. In this paper we compare the performance of IRDT and LPL by computer simulation. In particular, the long term operation of IRDT is examined by comparing power consumption under conditions of low data incidence. We also propose an improved IRDT and evaluate its performance. Improved IRDT dynamically sets the intermittent intervals to avoid performance degradation of the original IRDT.

II. OUTLINE OF IRDT

A. Intermittent Operation

Each node operating with IRDT sends its own ID intermittently in the network (Fig. 1(b)). Soon after sending the ID, its state becomes 'receive wait', and a little after that, it makes a transition to 'sleep' state. The sender node waits for the receiver node's ID in the 'receive wait' state and when the sender node gets an ID from an appropriate receiver, the sender returns a send request (SREQ) packet. After getting an acknowledge packet for the SREQ (RACK), the sender transmits a data packet and finishes communication following receipt of an acknowledge packet for the data (DACK).

Intermittent operation of node 1 and 2 is shown in Fig. 1(b). Node 3 is the sender and checks the ID from node 2 and ignores it because node 2 is an inappropriate receiver for node 3: node 3 accepts node 1 as an appropriate receiver. Then, the appropriate or inappropriate receiver is determined by the routing protocol (Section II-B). The sender node has one or more communication candidates, which improves communication reliability and reduces the sender's active time waiting for receivers to wake up.

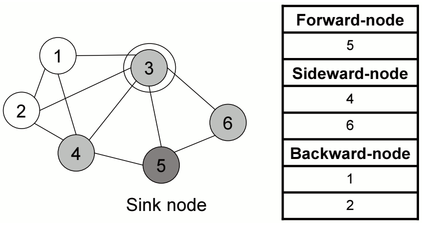


Fig. 2. Classification of neighbor nodes at node 3

B. Routing Protocol

The routing algorithm of IRDT is based on multi-hop routing, so each node conducts the relay processing of the packet. For reasons of energy consumption, minimum hop routing is preferable, but in some situations nodes cannot do so due to bad radio wave conditions or failure in certain nodes. Therefore, for higher flexibility, IRDT's routing algorithm considers alternative paths to the minimum hop route. All nodes have a configuration table managing topology information. They update their own tables by periodically interchanging topology information packets and they use their own tables to determine the number of hops from the sink node to themselves. If the minimum number of hops from one node to the sink node is denoted as H, the number of hops of its neighbor nodes is $H-1$, H, or $H+1$ and we call them the forward-node, sideward-node, or backward-node, respectively. For example, in Fig 2, node 3's neighbors are classified.

For minimum hop routing, the sender node prefers to select forward-nodes as receivers. When the sender gets a forward-node's ID, it returns a SREQ packet. We define communication failure as a situation in which the sender cannot get a RACK and DACK from the receiver. Sideward-nodes are selected when communication failures have occurred with all forward-nodes, and backward-nodes are selected if the same situation occurs for all sideward-nodes. All data packets have a time to live (TTL) field to avoid heavy repetitions of data relay. For each relay of a data packet, TTL is decremented by one and when TTL is 0 then the data packet is discarded. Any node will not select a sideward-node or a backward-node if this results in data packet loss due to the TTL mechanism.

C. Problem and Resolution of IRDT

The problem with IRDT is the collision of a SREQ packet with another SREQ packet. As described in Section II-B, the sender node returns a SREQ packet when an ID from a forward-node arrives. Thus, if the forward-node for more than one sender node sends an ID, the sender nodes simultaneously receive the IDs and return SREQs, so multi-SREQs collide with each other. In this case, because the sink node is the only forward-node for its neighbor nodes, the SREQ collision repeats. Because all nodes have a timer that is scheduled for discarding data, the repeated SREQ collisions will eventually stop. That timer is set to T_d, so the sender is in an idle state during T_d, which leads to large power consumption. We resolve this problem by changing the intermittent interval and randomly disregarding the forward-node.

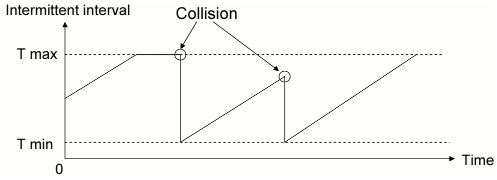


Fig. 3. Adaptive intermittent interval

SREQ collisions are caused by two factors. One factor is the disagreement between the transmission capacity and the load of a node. The maximum number of packets that each node can receive per unit time corresponds to the number of IDs each node sends per unit time. Therefore, as the intermittent interval of a node decreases, the amount of data that a node can receive increases. When the load surpasses the node's processing performance, multi-SREQs are sent and collide. Accordingly, each node dynamically and adaptively sets its ID transmission interval. Nodes determine that their loads are high when they wait for SREQ and collisions are detected. Then, they set their own intermittent intervals to T_{min}. If a SREQ collision is not detected, the nodes gradually increase their intervals to T_{max} to reduce the duty cycle (Fig. 3). T_{max} and T_{min} require careful configuration. A longer T_{max} decreases the node's own duty cycle, but increases its neighbors' idling time waiting for an ID, and a shorter T_{min} improves transfer performance, but obstructs the communication of other nodes.

The other factor is the priority of forward-nodes for the receiver. Therefore, if more than one node wants to send data to the same receiver, whenever the receiver transmits an ID, a SREQ collision occurs. At a node that is the only forward-node for many sender nodes, such as the sink node, the SREQ collision repeatedly occurs. To solve this problem, it is necessary that nodes ignore the ID of their forward-nodes randomly. Therefore, only when all transmissions to forward-nodes have failed, IDs from the forward-node are ignored in P_{reject} because constant avoiding of the forward-node leads to an increase of a receiver node's own idling time and the number of data relays. This additional process cannot prevent the first collision, but once a collision occurs, each sender node autonomously avoids continuous collisions.

III. SIMULATION RESULTS

In this section, we evaluate and compare the performance of IRDT and LPL using computer simulation. In IRDT, the number of communication candidates for each node would influence their performance. Here we use the topology in Fig. 4 where many nodes generally have several forward-nodes distributed within the observation area and the communication range of nodes is 100 m. In this topology, the sink node is in the bottom-left corner (represented by square) and other the 49 nodes (represented by circles, triangles, and diamond shapes) transmit the packets to the sink node randomly in accordance with the provided frequency. In Fig. 4, the 49 nodes are 1 to 8 hops away from the sink node and the same shape and color means the same number of hops. In detail, the average number of forward-nodes of each node is 2.74. When modeling the network, we used the following assumptions.

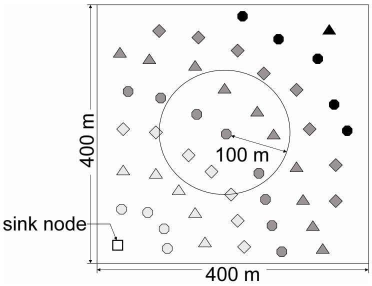


Fig. 4. Network model

TABLE I
PARAMETER SETTINGS

Parameter	Value
T_d	Intermittent interval$\times 5$
R_{limit}	$H+3$
Sending current	20 mA
Waiting current	25 mA
Sleeping current	0 mA
Packet size	128 byte
Transmission speed	100 kbps

- The network topology must not change during the simulation, i.e., each node does not exchange and update topology information, as mentioned in the previous section, and each node uses the configuration table that is determined before the start of the simulation.
- Each node conducts CSMA/CA to avoid collisions with other packets. Moreover, the strength of the radio signals does not decay, and unless packet collisions occur, a transmitted packet is assumed to be received by the nodes within the provided communication range.
- When collisions with other packets occur while a packet is being received, the packet is always discarded.

In our simulations, we investigated the packet collection ratio and the power consumption of each node when the parameters are set as shown in Table I and the packet generation rate (the number of data packet generated at each node per 1 s) is changed. Simulation time is 6 hours in all results. We define the packet collection ratio as the value of the number of packets received at the sink node divided by the number of all generated packets. We use the word 'performance' to indicate the packet collection ratio and the power consumption. The routing algorithm of IRDT is described in Section II-B, and the R_{limit} is the TTL. We assume a low packet generation rate and few collisions, meaning that relays to forward-nodes are mainly selected; hence, TTL is set to $H+3$ (H is the number of hops from the sink node) because extra relays increase power consumption. T_d (explained in Section II-C) is five times larger than an intermittent interval, so each node can get at least five times an ID from a forward-node from the start to end of the timer. LPL's routing algorithm is the same as that in Ref. [6], although because of the above-mentioned modeling assumption, each node would select only one of the forward-nodes that has the most remaining energy.

A. Basic Performance

The performance of both methods is examined for the topology shown in Fig. 4. For investigating the basic performance, the intermittent interval of all nodes is equally set to a constant value. Shorter intermittent intervals are already

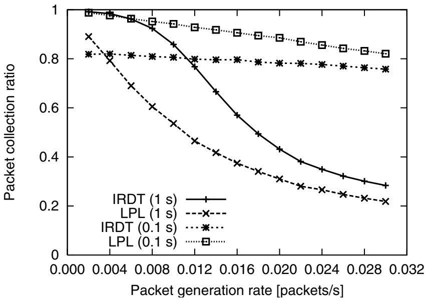


Fig. 5. Packet collection ratio

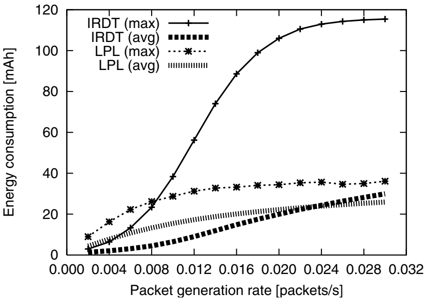
known to be important for improving the performance in IRDT from our previous research, but extremely shortened intervals cause frequent transmission of IDs, which seems to obstruct other communication. Therefore, we examine the basic performances when an intermittent interval is set to 0.1 or 1 s and clarify the performance characteristics of IRDT by comparing them with those of LPL.

1) Packet Collection Ratio: The collection ratio of both methods is shown in Fig. 5. In case that the intermittent interval is set to 1 s, IRDT can attain a comparatively high collection ratio in the situation where a packet generation rate is low, but cannot attain even 40% collection ratio at comparative high packet generation rate. In contrast, when the intermittent interval is 0.1 s, IRDT can always obtain about 80% collection ratio. Note that the collection ratio in IRDT is suppressed to only 82% even at a low packet generation ratio because highly frequent ID transmissions obstruct the communication of other nodes. In LPL, however, the collection ratio is lower than that of IRDT at 1 s because a long intermittent interval needs a long preamble, which obstructs other communication. However, the collection ratio at 0.1 s is better than that of IRDT. In LPL, each node intermittently enters the ‘listen’ state, so a short interval does not obstruct communication like in IRDT. Therefore, LPL can make intermittent intervals much shorter and obtain a much higher collection ratio.

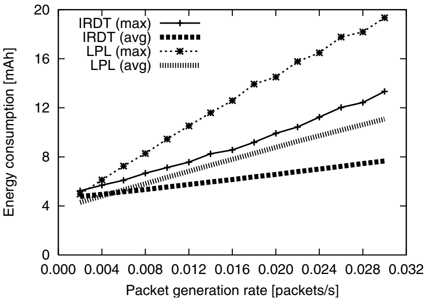
2) Energy Consumption: We examine the energy consumption characteristics of the highest loaded nodes and the average energy consumption between all nodes (Fig. 6). In a comparison of IRDT with LPL at a low packet generation rate when the intermittent interval is 1 s, IRDT’s power consumption is about the quarter of that of LPL due to IRDT being able to have more than one receiver.

In IRDT, more energy is consumed when sender nodes wait for receiver IDs, so the sink node consumes less energy than the other nodes do. In contrast, the neighbor nodes of the sink node consume large power because SREQ collisions occur more frequently at the sink node, which lengthen the time in the ‘receive wait’ (Fig. 6(a)). Thus, the energy consumption of the sink-neighbor nodes (IRDT (max)) grows rapidly in accordance with the increase of the packet generation rate when the intermittent interval is 1 s. Similarly, power consumption increases at the node whose receivers have frequent collisions of SREQ. Therefore, to reduce energy consumption in IRDT, preventing SREQ collisions is very important.

In LPL, the procedures for the collision avoidance apart from CSMA/CA are not used. So, a short intermittent interval is necessary to improve the collection ratio, but this lengthens the duty cycle. In IRDT, a longer interval is achieved at a low packet generation rate since nodes can use multiple paths. In contrast, when a packet generation rate is high, the packet collection ratio and the energy consumption in LPL are superior to those of IRDT due to SREQ collisions.



(a) Intermittent interval: 1 s



(b) Intermittent interval: 0.1 s

Fig. 6. Energy consumption

When intermittent interval is 0.1 s, the energy consumptions of both methods increase linearly, and the energy consumption of IRDT (max) doesn’t grow as 1 s because of less SREQ collisions (Fig. 6(b)). LPL consumes more energy than IRDT because nodes with LPL enter the ‘receive wait’ state even when they receive packets unrelated to own communication, which increases the duty cycle.

3) Basic Performance of IRDT: The advantages and disadvantages of setting an intermittent interval to improve the collection ratio are clarified in the previous section. When more packets are generated, the performance of IRDT is worse than LPL, when a short interval is used. However, when few packets are generated, IRDT can achieve a high collection ratio with low energy consumption in comparison with LPL. This deterioration in the performance of IRDT in accordance with an increasing packet generation rate is caused by SREQ collisions. Therefore, preventing SREQ collisions should improve the performance.

B. Dynamic and Adaptive Settings of the Intermittent Interval

Now we introduce the method for SREQ collision avoidance described in Section II-C to IRDT. We compare the performance of this improved IRDT with that of LPL when both methods dynamically set intermittent intervals which are adaptive to the load.

The setting of the intermittent interval is shown in Fig. 3 and parameters are shown in Table II. These parameters are experimentally-determined such that T_{max} is set in considera-

TABLE II
PARAMETER SETTINGS

Parameter	Value
T_{max}	1.5 s
T_{min}	0.2 s
T_n	$10 \times n$ ms
P_{reject}	50%

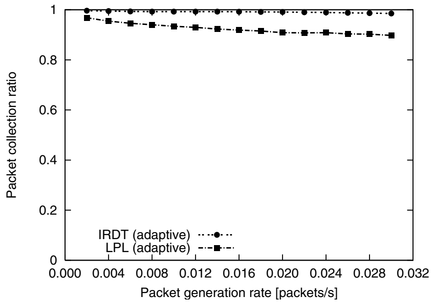


Fig. 7. Packet collection ratio

tion of the operation for about several years and short T_{min} is set to extent in which other communications are not obstructed. After the interval becomes T_{min}, the interval increases by T_n every transmission of an ID. The number n is the number of hops from the sink node, so a larger n means a lower load. Therefore, when n is large, T_n is also large in order to quickly return to T_{max} for duty cycle. In LPL, each node changes its own interval in accordance with the number of the data packets it sends during a time period. This adaptive LPL's intermittent interval is selected from 8 values (10, 20, 50, 100, 200, 400, 800, 1600 ms) because of its MAC layer protocol, BMAC [4].

In the improved IRDT, the collection ratio and energy consumption improve due to the adaptive settings of the intermittent interval. Especially, the collection ratio in improved IRDT is 98.5% even when the packet generation rate is 0.030 (Fig. 7). This result indicates that IRDT can perform effectively at a comparatively high packet generation rate by using adaptive interval settings. Because the parameters in adaptive LPL are aimed at managing power consumption, the collection ratio is lower than that of the improved IRDT. As for average energy consumption, when packets are generated infrequently, the improved IRDT obtain a 50 % reduction from adaptive LPL (Fig. 8), and the improved IRDT can obtain both, a higher collection ratio and a lower average energy consumption than those reported in Section III-A1. Although at low packet generation rate, the energy consumption of the highest loaded node, IRDT adaptive (max) in Fig. 8, is larger than the original IRDT's result with 1 s intermittent interval. Furthermore, at any packet generation rate, IRDT adaptive (max) is larger than LPL adaptive (max) which increases in a stair case pattern because of the discrete setting of its intermittent interval as mentioned above. This decreases the packet collection ratio because the limited batteries of the sink-neighbor nodes run out sooner. If SREQ collisions tend to occur in the sink-neighbor nodes, for example there are much more sink-neighbor nodes, improved IRDT is more effective even at sink-neighbor nodes than does the original. And to achieve a high enough collection ratio in adaptive LPL, a shorter intermittent interval is necessary, and this increases energy consumption because of the high duty cycle. In this regard, we need to investigate in our next research.

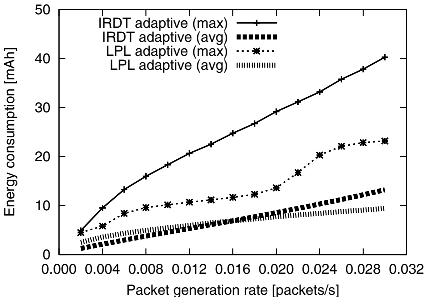


Fig. 8. Energy consumption

IV. CONCLUSION

In this paper, we studied the basic performance characteristics of the receiver-driven asynchronous system IRDT. Simulation results showed that under the conditions of a small data-generating frequency, the packet collection ratio of IRDT is as high as that of LPL and IRDT can cut the power consumption of LPL by 75%. We also improved IRDT in which each node dynamically sets its own intermittent interval adaptively to the load. A comparison of the improved IRDT and adaptive LPL showed that the packet collection ratio of improved IRDT is higher than that of adaptive LPL and the improved IRDT has 50% less power consumption than adaptive LPL at a low packet generation ratio. The efficient settings of T_{max} and T_{min} and a performance evaluation considering network life, a real environment and applications are intended as future work.

ACKNOWLEDGMENT

This research was supported in part by "Global COE (Centers of Excellence) Program" of the Ministry of Education, Culture, Sports, Science and Technology, Japan and "Grant-in-Aid for Scientific Research (A) 21240004" of the Japan Society for the Promotion of Science (JSPS) in Japan.

REFERENCES

[1] X. Du, Y. Xiao, and F. Dai, "Increasing network lifetime by balancing node energy consumption in heterogeous sensor networks," *Wireless Communications and Mobile Computing*, vol. 8, no. 1, pp. 125–136, Jan 2008.

[2] S. J. Baek and G. de Veciana, "Spatial energy balancing through proactive multipath routing in wireless multihop networks," *IEEE/ACM Transactions on Networking*, vol. 15, no. 1, pp. 93–104, Feb 2007.

[3] K. Pister and L. Doherty, "TSMP:Time Synchronized Mesh Protocol," in *Proceedings of the 20th IASTED International Conference on Parallel and Distributed Computing and Systems*, 2008.

[4] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*. ACM New York, NY, USA, 2004, pp. 95–107.

[5] M. Buettner, G. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems*. ACM Press New York, NY, USA, 2006, pp. 307–320.

[6] R. Jurdak, P. Baldi, and C. V. Lopes, "Adaptive low power listening for wireless sensor networks," *IEEE Trans. Mobile Computing*, vol. 6, no. 8, pp. 988–1004, Aug 2007.

[7] M. Sugano, R. Fukushima, M. Murata, T. Hayashi, and T. Hatauchi, "Performance Evaluation of a Low-Energy-Consumption Ad Hoc Mesh Network Based on Intermittent Operation," in *The 3rd IEEE Workshop on Wireless Mesh Networks (WiMesh 2008, Poster session)*, Jun 2008.