Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2010, Article ID 169132, 12 pages
doi:10.1155/2010/169132

Research Article

A Study on Event-Driven TDMA Protocol for Wireless Sensor Networks

Haigang Gong,[1] Ming Liu,[1] Guihai Chen,[2] and Xue Zhang[1]

[1] *School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 6 10054, China*
[2] *Deparment of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200242, China*

Correspondence should be addressed to Haigang Gong, hggong@uestc.edu.cn

Received 14 February 2010; Accepted 3 November 2010

Academic Editor: Xiang-Yang Li

Copyright © 2010 Haigang Gong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

MAC protocol controls the activity of wireless radio of sensor nodes directly so that it is the major consumer of sensor energy and the energy efficiency of MAC protocol makes a strong impact on the network performance. TDMA-based MAC protocol is inherently collision-free and can rule out idle listening since nodes know when to transmit. However, conventional TDMA protocol is not suitable for event-driven applications. In this paper, we present ED-TDMA, an event-driven TDMA protocol for wireless sensor networks. Then we conduct extensive simulations to compare it with other MAC protocols such as BMA, S-MAC, and LMAC. Simulation results show that ED-TDMA performs better for event-driven application in wireless sensor networks with high-density deployment and under low traffic.

1. Introduction

Like in all other shared-medium networks, medium access control (MAC) is also a key component to ensure the successful operation of wireless sensor networks. A MAC protocol decides when competing nodes could access the shared medium and tries to ensure that no collisions occur while nodes' transmission. Compared to nodes in traditional wireless networks, the main constraint of sensor nodes in WSNs is their low finite battery energy. Since sensor nodes are often powered by battery and left unattended after deployment, for example, in hostile or hash environments, making it difficult to replace or recharge their batteries, MAC protocols running on WSN must consume energy-efficiently in order to achieve a longer network lifetime. According to Estrin et al. [1], the radio component of sensor nodes consumes most of nodes' energy when receiving or transmitting data, even in idle mode. On the other hand, medium access control (MAC) protocol directly controls the activity of nodes' radio and decides when the competing nodes may access the shared medium to transmit the data. So, medium access is the major consumer of nodes' energy and MAC protocols must be energy-efficient.

When running a MAC protocol, much energy is wasted due to the following sources of overhead: (a) *Idle listening*: since a node does not know when it will be the receiver of a message from one of its neighbors, it must keep its radio in idle listening mode at all times. (b) *Collisions*: if two nodes transmit at the same time and interfere with each other, collisions happen and packets are corrupted. (c) *Overhearing*: a node may receive packets that are not destined for it. In fact, it would have been more efficient to turn off its radio. (d) *Protocol overhead*: the MAC headers and control packets used for signaling do not contain application data and are therefore considered overhead.

MAC protocols designed for wireless sensor network can be broadly divided into schedule-based and contention-based protocols [2]. Schedule-based MAC protocols, including TDMA, FDMA and CDMA, have a central point permitting the access to the shared medium by broadcasting a schedule that specifies when each node may transmit over the shared medium. The lack of contention overhead guarantees that the method robust when traffic load is high. Furthermore, with the proper scheduling, nodes can get deterministic access to the medium and can provide delay-bounded services. For contention-based MAC protocols such as IEEE 802.11 [3], S-MAC [4], T-MAC [5], they must handle the possible collisions while data transmission. Contention-based MAC protocols may deal with collisions through some contention resolution scheme such as retransmitting the data later or occupying the shared medium before data transmission. Compared with schedule-based MAC protocols, contention-based MAC protocols consume more energy because they waste energy in collisions and idle listening. Moreover, they do not give delay guarantees. However, they are very flexible and can handle the traffic fluctuations in wireless sensor networks.

In schedule-based MAC protocols, TDMA is more power efficient because it is inherently collision-free and can avoid unnecessary idle listening. For example, the TDMA protocol for a traffic-monitoring network described in [6] has a lifetime of 1,200 days compared with ten days using the IEEE 802.11 protocol. For the inherently property of energy conserving, TDMA protocols have been recently attracted significant attention for many applications [7–10].

However, TDMA only applies for continuous monitoring applications, that is, continuous collecting the temperature or humidity of the environments. They could achieve high channel utility because sensor nodes always have data to send in continuous data gathering applications. But when applying for another typical application in WSNs-event-driven applications such as earthquake monitoring or target tracking, in which sensor nodes only have data to send when a specific event occurs, they will waste more energy and achieve lower channel utility for that sensor nodes still must be active when the event does not happen.

In this paper, we present ED-TDMA, an event-driven TDMA protocol for wireless sensor networks. And extensive simulations are conducted to compare it with other MAC protocol such as BMA [10], S-MAC and LMAC [11] in different scenarios. Simulation results show that ED-TDMA performs better for wireless sensor network with high-density deployment and low traffic.

The rest of the paper is organized as follows. Section 2 discusses some typical MAC protocols. Section 3 presents the problem and system model. Section 4 describes our ED-TDMA protocol in detail and analyzes its energy consumption. Simulation results are discussed in Section 5. Finally, Section 6 concludes the paper.

2. Related Works

2.1. Contention-Based MAC Protocols. Sensor-MAC (S-MAC) protocol [4] is a contention-based effective MAC protocol designed by Ye et al. for wireless sensor networks The basic idea of S-MAC is that time is divided into large frames. Every frame starts off with a small synchronization phase, followed by a fixed active part and a sleep part. During synchronization phase, nodes receive or send SYNC packet contained the schedule information (i.e., when to sleep). During the sleep part, a node turns off its radio to preserve energy. During the active part, it can communicate with its neighbors and send any messages queued during the sleep part. Since all messages are packed into the active part, instead of the whole frame, therefore the energy wasted on idle listening is reduced.

Timeout-MAC (T-MAC) protocol [5] introduces an adaptive duty cycle too. In T-MAC, a node keeps listening and potentially transmitting as long as it is in an active period. If a node does not detect any activity within the time-out interval, it can safely assume that no neighbor wants to communicate with it and goes to sleep. The activation time events include reception of any data, the sensing of communication on the radio, and so forth. Simulations show that T-MAC gives better results under different loads. However, T-MAC breaks the synchronization of the listen periods, and introduces early sleep problem which is harmful to the network performance.

TA-MAC [12] modifies the contention window mechanism of S-MAC. It adjusts the initial contention window according to the current traffic load to reduce the collision probability and employs a fast back-off scheme to reduce the time for idle listening during back-off procedure, which reducing the energy consumption. Simulation results have shown that TA-MAC achieves energy savings and higher throughput when traffic load is heavy.

2.2. Schedule-Based MAC Protocols. In schedule-based MAC protocols, TDMA is inherently collision-free and can avoid unnecessary idle listening. The main task in TDMA scheduling is to allocate time slots depending on the network topology and the node packet generation rates. A proper schedule not only avoids collisions by silencing the interferers of every receiver node in each time slot but also minimizes the number of time slots hence the latency. TDMA protocols could be categorized into cluster-based TDMA and distributed TDMA. The former are for networks in which the nodes are organized into several clusters, and cluster heads allocate time slots to their members. Distributed TDMA is more challenging than cluster-based TDMA because spatial reuse of a time slot may be possible. More than one node can transmit at the same time slot if their receivers are at nonconflicting parts of the network.

BMA protocol is a cluster-based protocol which improves traditional TDMA schedule in that there exists a contention phase (CP) in the beginning of each TDMA frame. In the contention phase during each frame, source nodes send 1-bit message to their cluster heads to reserve time slot so that cluster heads know which members will transmit in this frame and allocate successive time slot to these source nodes. When the source nodes finish their transmission, cluster heads could be asleep and will be active in the next frame. While saving energy for sleeping after transmission, BMA introduces extra schedule overheads for its TDMA scheduling. Moreover, it achieves poor channel utility for event-driven applications.

On the other hand, distributed TDMA is more complex than cluster-based TDMA because it must allocate nonconflicting time slots to all the nodes in the network. That is to say, two or more nodes can transmit simultaneously if their receivers are at nonconflicting parts of the network. Obviously, it is not an easy task.

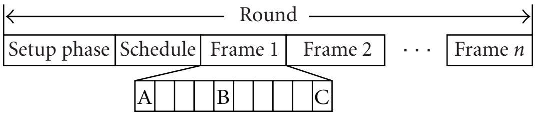

Figure 1: Frame structure of traditional TDMA protocol.

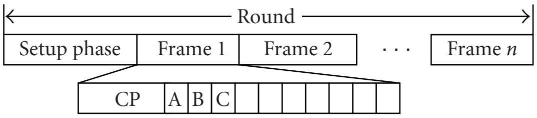

Figure 2: Frame structure of BMA protocol.

LMAC [11] is a typical distributed TDMA protocol. Nodes organize time into slots, grouped into fixed-length frames. A slot consists of a traffic control section and a fixed-length data section. The scheduling discipline is extremely simple: each active node is in control of a slot. When a node wants to send a packet, it broadcasts a message header in the control section detailing the destination and length until its time-slot comes around, and then immediately proceeds with transmitting the data. Nodes listening to the control header turn off their radio during the data part if they are not an intended receiver of the message. However, nodes must always listen to the control sections of all slots in a frame, even if the slots are unused.

TRAMA protocol [13] is another distributed TDMA protocol. Nodes periodically exchange their information and learn their two-hop neighborhood. Based on this knowledge, nodes periodically reserve future slots for backlogged traffic. A hash-based priority scheme is then used so that only one node in a two-hop neighborhood will transmit in a given slot. Unfortunately, the TRAMA protocol implementation is complex and assumes application-level forecasting of traffic.

Z-MAC [14] is a hybrid protocol, focusing on recapturing wasted slots by allowing nodes to compete for all slots with a bias towards the owner of the slot. This method allows nodes to recapture unused bandwidth without having to renegotiate the slot schedule. However, it removes the collision-free guarantee on message transmission and often cannot fully recover the bandwidth. It also does not solve the problem of requiring time synchronization amongst communicating nodes.

3. Problem Statement and System Model

3.1. Problem Statement. As mentioned before, traditional TDMA schedule is effective for continuous monitoring application while nodes have the data to send all the time. But for event-driven application, it has some disadvantages such as lower channel utility and unnecessary energy wastage of the cluster heads. HEED [15] is a clustering protocol integrating with traditional TDMA schedule. The operation of HEED is divided into rounds. As shown in Figure 1, each round begins with a set-up phase, followed by a TDMA schedule phase and several TDMA frames. In the set-up phase, sensor nodes are organized into several clusters. And then the cluster heads broadcast a TDMA schedule to their members, allocating a slot to the members. In the following TDMA frames, the members send the data to their respective cluster heads during the allocated slot. There is only 1 TDMA schedule in each round and the length of TDMA frame is equal. As in Figure 1, TDMA frame contains 10 slots. If there are only several source nodes to transmit during a frame, there must be some empty slots. For example, node A, B, and C transmit their data during the first, the fifth and the tenth slot, respectively, then 7 slots are empty which wastes network bandwidth and decreases the channel utility. Moreover, cluster heads do not know which members will send their data in the current TDMA frame so that cluster heads must be active during the round even if there have no data to transmit, which leads to unnecessary energy wastage of cluster heads.

Figure 2 shows the frame structure of BMA, which improving traditional TDMA schedule by inserting a contention phase in the beginning of each frame. As in Figure 2, source node A, B, and C transmit during the first three data slots and their cluster head could enter into sleep state in the forth data slot to avoid unnecessary energy wastage. However, like in traditional TDMA protocol, TDMA frames in BMA protocol have the same length, which couldnot improve channel utility of the network. Once an event occurs, the sensors related to the event will send the sensing data during a period of time. If the length of TDMA frame is constant, then the sensors must send their data in the next frame even if there have some empty slots in the current frame. In addition, there's a TDMA schedule in each frame and cluster heads will broadcast a TDMA schedule packet in each frame. The schedule packet includes the member's ID and the slot number allocating to the members, which introduces extra energy overhead. Broadcasting and receiving these schedule packets consume considerable energy when the node density is high.

Our ED-TDMA protocol then improves channel utility by changing the length of TDMA frame according to the number of source nodes and reduces the length of TDMA schedule packets with a bitmap-assisted TDMA schedule to decrease the schedule overhead. Besides, it employs intracluster coverage scheme to save nodes' energy so as to prolong network lifetime and to improve system scalability.

3.2. System Model. Assume that N nodes are dispersed in a square $L \times L$ field randomly, and the follow assumptions hold:

(1) The only base station sits at a fixed location outside the field.

(2) Power control is available. Intracluster and intercluster communication use different power level.

(3) All nodes have same capabilities and data fusion is capable.

(4) Nodes are left unattended after deployment and nodes are stationary.

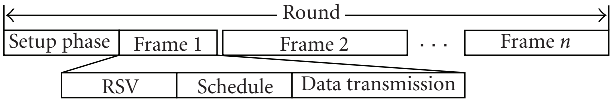

FIGURE 3: Frame structure of ED-TDMA.

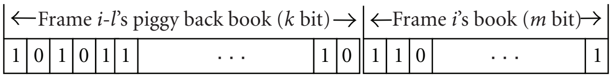

FIGURE 4: TDMA schedule packet.

After deployment, nodes are partition into several clusters and cluster heads are organized into a routing tree. Cluster heads assign time slots to the source nodes in their clusters. The source nodes send their data to cluster heads that relay the data along the routing tree. Finally, the root node transmits the aggregated data to the base station. Clustering and routing tree building are beyond the scope of this paper.

Besides, we use the same radio model in [16] for the radio hardware energy dissipation where the transmitter dissipates energy to run the radio electronics and the power amplifier, and the receiver dissipates energy to run the radio electronics. To transmit a k bit message a distance d, the radio expends energy as follows:

$$E_{\mathrm{Tx}} = \begin{cases} k * E_{\mathrm{elec}} + k * e_{\mathrm{fs}} d^2, & d < d_0, \\ k * E_{\mathrm{elec}} + k * e_{\mathrm{amp}} d^4, & d \geq d_0. \end{cases} \tag{1}$$

And to receive this message, the radio expends energy as follows:

$$E_{\mathrm{Rx}} = k * E_{\mathrm{elec}}. \tag{2}$$

E_{elec}, the electronics energy, depends on factors such as the digital coding, modulation, and filtering of the signal before it is sent to the transmit amplifier. And the amplifier energy, $e_{\mathrm{fs}} d^2$ or $e_{\mathrm{amp}} d^4$, depends on the distance to the receiver.

4. ED-TDMA Protocol Design

4.1. Basic Protocol. Like BMA, the operation of ED-TDMA is divided into rounds. Each round begins with a set-up phase, followed by a steady phase. Set-up phase includes clustering and time synchronization. The steady phase consists of n variable-length TDMA frames. As shown in Figure 3, each frame begins with a reservation phase, followed by a TDMA schedule and data transmission.

The reservation phase consists of m mini-slot. m is the number of members in the cluster. The members occupy the mini-slot according to their ID. Node having the maximum ID occupies the first mini-slot while node having the minimum ID occupies the last mini-slot, and so on. A member sends a 1-bit RSV message to the cluster head if it has data to send in the current frame. Obviously, the length of the reservation phase is m bit.

In the TDMA schedule phase, the cluster head broadcasts a schedule packet according to the received RSV message in the reservation phase. The schedule packet format is a bit-map sequence as shown in Figure 4. The sequence consists of two parts. The first k bit part represents the piggybacking reservation of the previous frame, in which each bit corresponds to a source node in the previous frame. The second m bit part represents the reservation of the current frame, in which each bit corresponds to a node in the current frame. The piggybacking reservation has preference to the current reservation. Parameter k represents the number of the source nodes or the number of time slots in the previous frame and it satisfies $0 \leq k \leq m$. The value of k is variable with the number of the source nodes and is set to 0 in the first frame of a round. In the schedule sequence, 1 means a source node has booked a time slot. If a source node reserves time slot in the ith mini-slot, then it corresponds to the ith bit of the last m bit of the schedule sequence. If a source node reserves time slot by piggybacking reservation and it transmits data during the jth data time in the previous frame, it is reservation corresponds the jth bit of the first k bit of the schedule sequence. A source node determines its time slot number according to the number of bits 1 in the substring of the schedule sequence ending at its corresponding bit. Obviously, the number of bits 1 is the number of time slots, k, in the current frame. All members in the cluster, including source nodes and nonsource nodes, could get the knowledge of k from the schedule packet and then enter the reservation phase of the next frame after k time slots. If the number of source nodes is small, the frame length is too short which introduces frequent reservation and TDMA schedule, leading to more energy overhead. To avoid frequent reservation and schedule, when the number of source nodes is very small, we define a default minimum frame length $T_{\text{frame-min}}$. If the current frame length is less than $T_{\text{frame-min}}$, the frame length is set to $T_{\text{frame-min}}$.

For example, assuming that 4 source nodes A ∼ D send the RSV message to the cluster head in the 1st, 2nd, 4th, and mth mini-slot, respectively. The cluster head then broadcasts the TDMA schedule packet. In the schedule sequence shown in Figure 5, node A ∼ D correspond the 1st, 2nd, 4th, and mth bit of the schedule sequence, respectively. Note that the sequence has only the second m bit part in the first frame. The corresponding substring of node A is 1, then the slot number of node A is 1; the corresponding substring of C is 1101, then C occupies the 3rd time slot because the number of bits 1 in its substring is 3. Likewise, node B and node D occupy the 2nd and 4th data slot. From the sequence, all members in the cluster know that the first TDMA frame has 4 time slots. After 4 time slots, all members will enter into the second frame.

In the second frame, assuming that node A, C, D, E, and F have data to send. Then node E, and node F send the RSV message in the 3rd and 5th mini-slot in the reservation phase, assuming that node A, node C and node D reserve their time slot in the first frame by piggybacking. The TDMA schedule packet then contains two parts: the first 4 bit is the piggybacking reservation and the last m bit is the reservation of the current frame, as shown in Figure 6.

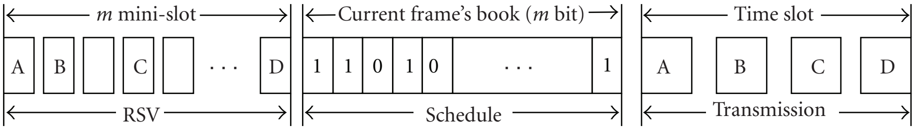

Figure 5: The first frame structure of ED-TDMA.

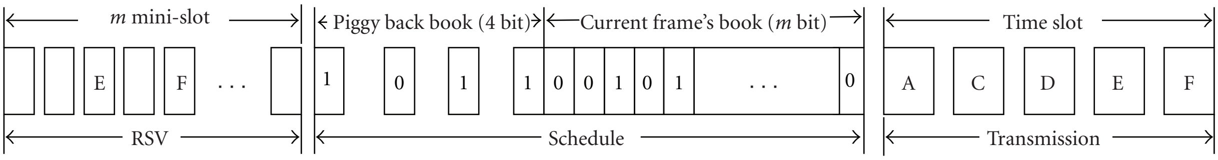

Figure 6: The second frame structure of ED-TDMA.

In the schedule sequence, node A, node C and node D correspond the 1st, 3rd and 4th bit of the sequence while node E and F correspond the 3rd and 5th bit of the last m bit of the sequence. Then the substring of node A, node C and node D are 1, 101 and 1011, respectively, which means the three nodes occupy the first 3 time slot in the second frame. Similarly, the corresponding substring of E is 1011001 so that the slot number of E is 4, and the corresponding substring of F is 101100101 so that node F occupies the 5th time slot. So, the data transmission phase is 5 data slots in the second frame.

In the transmission phase, the source nodes transmit the data to the cluster heads during its time slot. If they have more data to send in the next frame, they could book time slots of the next frame by piggybacking a flag in the data packet.

Noticeably, if there have no data to send, all nodes should be asleep for a default frame length to avoid frequent reservation and schedule. $T_{\text{frame-def}}$ is related to specific application. $T_{\text{frame-def}}$ could be longer if the application has no real time requirements.

Obviously, the length of the schedule packet is $(k + m)/8$ bytes. With $0 \leq k \leq m$, the length of the schedule packet, l_s, satisfies $m/8 \leq l_s \leq m/4$. For BMA and traditional TDMA, the length of the schedule packet, l'_s, is related to the number of the cluster members, m. Assuming that the schedule information includes the node's ID (2 bytes) and the slot number (1byte), then l'_s is $3m$ bytes.

The time of a round is predetermined and remains constant in the runtime, but the number of TDMA frames of the clusters in a round is different from each other because the number of source nodes in each cluster is different. In order to enter into the next round at the same time, cluster heads are responsible for determine an appropriate length of the last frame. For example, 7 nodes request for data transmitting in the last frame, but the network will enter into the next round after 4 data slot time. Then the cluster heads will notify the members that there are 4 data slots in the last frame. That is to say, only 4 source nodes would get its data slot number. The other 3 nodes will transmit their data during the first frame in the next round.

4.2. IntraCluster Coverage.

Coverage is one of the most important issues in WSNs and has been studied in recent years [17–19]. In most case, "coverage" means area coverage. And K-coverage can be descried as that every point in the monitored field is covered by at least K sensor. In [19], authors think it is hard to guarantee full coverage for a given randomly deployment area even if all sensors are on-duty. Small sensing holes are not likely to influence the effectiveness of sensor networks and are acceptable for most application scenarios. It is enough to meet the application's requirements if the active nodes in the network could maintain reasonable area coverage—coverage expectation. Coverage mechanism is to choose a subset of active nodes to maintain the coverage expectation.

We introduce this idea into clusters, that is, called "intracluster coverage," which selects some active node within clusters while maintaining coverage expectation of the cluster. Based on our previous work [20], cluster heads randomly choose m' active nodes according to the following:

$$P_{\text{cover}} = \sum_{i=K}^{m'} C_{m'}^{i} \left(\frac{r}{R}\right)^{2i} \left(1 - \frac{r^2}{R^2}\right)^{m'-i}, \tag{3}$$

where P_{cover} is the coverage expectation of sensing field determined by specific applications; and r is sensing radius, R is cluster radius; m' is the number of active nodes. For example, distributing 200 nodes in a $100 \times 100\,\text{m}^2$ field, $r = 12\,\text{m}$, $R = 30\,\text{m}$, then the average number of cluster members is 60 or so. With intracluster coverage, if $P_{\text{cover}} = 99\%$ which means 99% of sensing field is expected to be monitored, 27 members should be active in each cluster to ensure 1-coverage of the cluster and 38 members to ensure 2-coverage. If $P_{\text{cover}} = 95\%$, only 16 nodes and 25 nodes should be active to ensure 1-coverage and 2-coverage, respectively.

Using intracluster coverage has two advantages. The first advantage is to preserve energy consumption in each round by turning redundant nodes' radio off so that network lifetime is prolonged. The second is to reduce TDMA schedule overhead. Once clusters grouped, all cluster head broadcast a TDMA schedule packet in which contains the members' ID and slot number allocated to the members.

When node density is high, the number of cluster members turns higher so that the length of TDMA schedule packet gets longer that consumes more energy to transmit and receive. However, the length of TDMA schedule packet would not be too long with intracluster coverage because the number of active nodes varies slightly when node density goes higher. As in Table 1, the number of active nodes increases while the number of nodes increasing. If node density is high enough, the number of active nodes maintains a const.

4.3. Energy Analysis. Assume that there are m nodes and m_s source nodes in each cluster and the event whether a node has data to send or not can be viewed as a Bernoulli process, in which the probability that a node has data to send is p and there is $m_s = mp$.

To ED-TDMA, the source nodes' energy is consumed for sending the RSV message in the reservation phase, receiving TDMA schedule packet and transmitting data to the cluster head. It could be expressed as

$$\begin{aligned} E_s &= E_t(l_r, d_i) + E_r(l_s) + E(l_d, d_i) \\ &= (l_r + l_s + l_d)E_{\text{elec}} + (l_r + l_d)e_{\text{fs}}d_i^2, \end{aligned} \tag{4}$$

where d_i is the distance from source nodes to cluster head; l_r, l_s and l_d are the length of the reservation message, TDMA schedule packet and data packet, respectively.

Nonsource nodes consume energy only for receiving TDMA schedule packet

$$E_{\text{ns}} = E_r(l_s) = l_s E_{\text{elec}}. \tag{5}$$

E_{CH} is the energy consumption of the cluster head, including listening or receiving in the reservation phase, broadcasting TDMA schedule packet and receiving data packet from the source nodes

$$E_{\text{CH}} = mE_r(l_r) + E_t(l_s, r) + \sum_{i=1}^{m_s} E_r(l_d). \tag{6}$$

Then the total energy dissipated in a frame is

$$\begin{aligned} E_{\text{ED-TDMA}} &= E_{\text{CH}} + \sum_{i=1}^{m_s} E_s + (m - m_s)E_{\text{ns}} \\ &= [(m + m_s)l_r + (m + 1)l_s + 2m_s l_d]E_{\text{elec}} \\ &\quad + \sum_{i=1}^{m_s}(l_r + l_d)e_{\text{fs}}d_i^2 + l_s e_{\text{fs}}r^2. \end{aligned} \tag{7}$$

According to [8], the total energy consumption of BMA in a frame is

$$\begin{aligned} E_{\text{BMA}} &= [m(m + 1)l_r + (m + 1)l'_s + 2m_s l_d]E_{\text{elec}} \\ &\quad + \sum_{i=1}^{m_s}(l_r + l_d)e_{\text{fs}}{d_i}^2 + l'_s e_{\text{fs}}r^2. \end{aligned} \tag{8}$$

For traditional TDMA protocol, the energy is consumed for staying active during the frame and receiving data from the source nodes. It could be expressed as:

$$E_{\text{TDMA}} = (m + m_s)l_d E_{\text{elec}} + \sum_{i=1}^{m_s} l_d e_{\text{fs}} d_i^2. \tag{9}$$

Table 1: Relationship between the number of nodes and the number of active nodes ($100 \times 100\,\text{m}^2$, $P_{\text{cover}} = 95\%$).

The number of nodes	The number of active nodes
50	10
100	13
150	16
200	17
250	17
300	17

The length of the reservation message l_s is only 1 bit. And there are $m/8 \le l_s \le m/4$ and $l'_s = 3m$. Then we have

$$\begin{aligned} E_{\text{ED-TDMA}} - E_{\text{BMA}} &\le -(23m^2 + 22m - mp)E_{\text{elec}} - 22me_{\text{fs}}r^2 \\ &\le -(23m^2 + 21m)E_{\text{elec}} - 22me_{\text{fs}}r^2 \le 0. \end{aligned} \tag{10}$$

From (10), the larger m is, the less energy consumption of $E_{\text{ED-TDMA}}$ than that of E_{BMA}.

Besides, there is

$$\begin{aligned} &E_{\text{ED-TDMA}} - E_{\text{TDMA}} \\ &\quad \le [2m^2 + 10m + 8mp - m(1 - p)l_d]E_{\text{elec}} + 2me_{\text{fs}}r^2. \end{aligned} \tag{11}$$

It means that the relationship between $E_{\text{ED-TDMA}}$ and E_{TDMA} is related to the length of data packet, l_d.

5. Performance Evaluation

To evaluate the performance of ED-TDMA, we first compare it with BMA protocol and traditional TDMA in order to show that TDMA schedule and data transmission of ED-TDMA is more efficient than others. Then we make comparisons between ED-TDMA and other MAC protocols such as contention-based MAC protocol—S-MAC and distributed TDMA protocol—LMAC in different scenarios.

5.1. Simulation I

5.1.1. Experiment Setup. We implemented ED-TDMA, ED-TDMA1, BMA and traditional TDMA protocols in the glomosim network simulator with the wireless extension, in which ED-TDMA1 is the extension of the basic ED-TDMA with intracluster coverage scheme. Simulation parameters are listed in Table 2. Assuming that data transfer rate is 19.2 kbps, which is the data transfer rate of TR1000 [21] when using OOK modulation, then transmitting 100 bytes data needs 42 ms. A time slot is set to 45 ms, which is long enough to send 100 bytes data to the cluster head. For ED-TDMA, $T_{\text{frame-min}}$ is relevant to sampling frequency and sampling resolution of the sensors and should be long enough to generate a data packet. When data is sampled at 100 Hz and 16 bits per sample, $T_{\text{frame-min}}$ is set to 495 ms. The reservation phase and schedule phase could be accomplished in a time slot. Set $T_{\text{rsv}} + T_{\text{schedule}} = 45\,\text{ms}$. Moreover,

Table 2: Simulation I parameters

Parameters	Value
Sensor area ($L \times L$)	$100 \times 100\,\text{m}^2$
The number of nodes (N)	300
Sensing radius (r)	12 m
Cluster radius (R)	30 m
E_{elec}	50 nJ/bit
e_{amp}	$0.0013\,\text{pJ/bit/m}^4$
e_{fs}	$10\,\text{pJ/bit/m}^2$
Initial energy	2.0 J
P_{cover}	95%
n	10
T_{slot}	45 ms
$T_{\text{frame-min}}$	495 ms
$T_{\text{frame-def}}$	9.9 s
$T_{\text{rsv}} + T_{\text{schedule}}$	45 ms

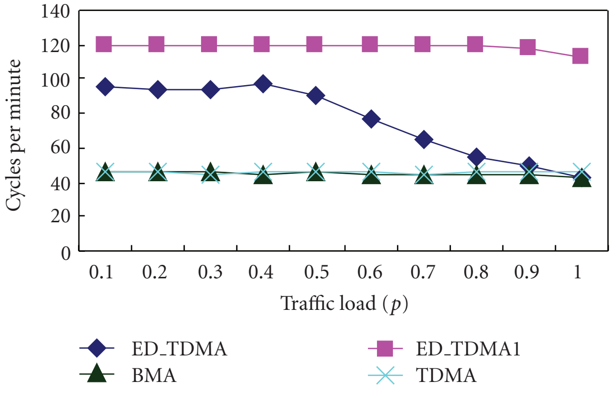

Figure 7: Cycles per minute under different load.

we assume that the packets are generated according to Bernoulli process. The transmission probability is p, which controls the network load.

5.1.2. Simulation Results. Figures 7 and 8 show the data cycles per minute and the transmitted the number of packets under different network traffic load. A data cycle corresponds to a TDMA frame, which all clusters collect data from their members in a frame. Obviously, the data cycles of ED-TDMA are almost twice more than that of BMA and traditional TDMA when traffic load is low. The reason is that the length of ED-TDMA varies with the number of source nodes so that its TDMA frame is shorter than the other two when traffic load is light. Therefore, ED-TDMA could perform more data cycles and transmit more data packet than BMA and traditional TDMA in the same period. With the increasing of traffic load, the length of TDMA frame of ED-TDMA increases so that the data cycles decrease and are nearly the same as BMA when all nodes have data to send ($p = 1$). In addition, ED-TDMA1 performs better than ED-TDMA because the intracluster coverage scheme ensures that frame length of ED-TDMA1 remains constant under different traffic load.

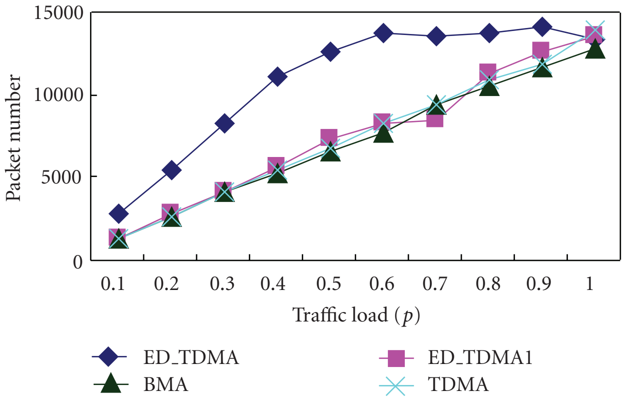

Figure 8: Packet received under different load.

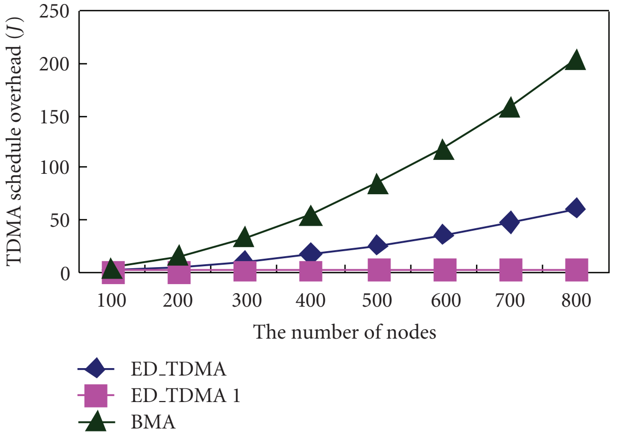

Figure 9: TDMA schedule overhead versus the number of nodes.

Figures 9 and 10 plot the TDMA schedule overheads after 5000 data cycles under different node density and different traffic load, respectively. With the increase of the node density, which means the number of members in the cluster increases, the schedule overhead of BMA increases rapidly and is far more than ED-TDMA. For example, when node density is $0.04\,\text{nodes/m}^2$, the schedule overhead of BMA is triple than ED-TDMA. When node density is constant and the traffic load turns higher, the number of source nodes increases which increases the length of the schedule packet so that schedule overhead also increases. For ED-TDMA, the max length of schedule packet is 2m bits so that its schedule overhead increases slowly. For ED-TDMA1, the number of working nodes is constant and is far less than others; its schedule overhead is very small and is independent on the node density and traffic load.

Figures 11 and 12 show the energy consumption after 5000 data cycles under different node density and different traffic load. Obviously, there are more energy consumptions with the increase of node density or traffic load. And BMA consumes energy more quickly than ED-TDMA. For instance, BMA consumes about 25% more energy than ED-TDMA and about 91% more than ED-TDMA1,

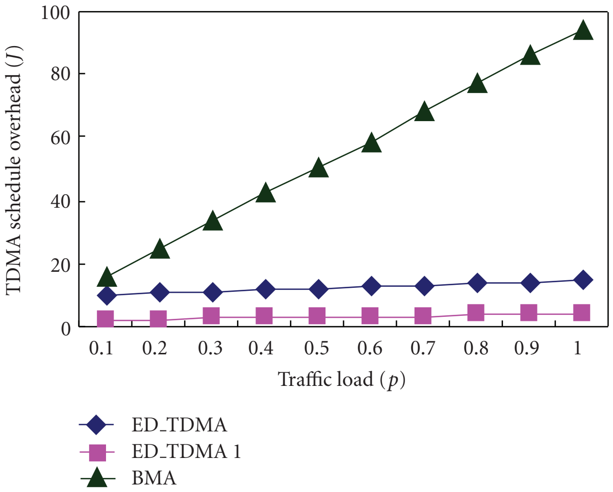

FIGURE 10: TDMA schedule overhead versus traffic load.

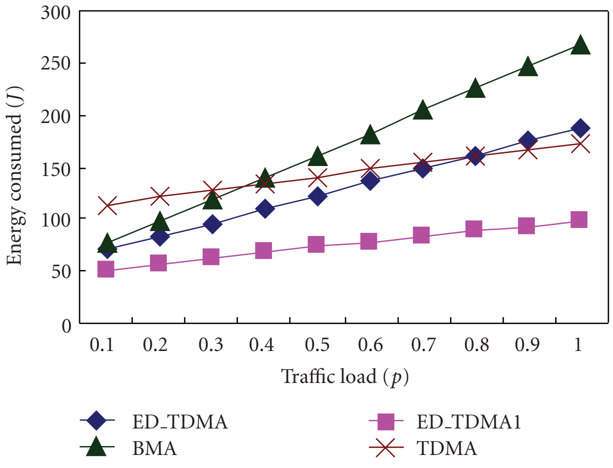

FIGURE 12: Energy consumption versus traffic load.

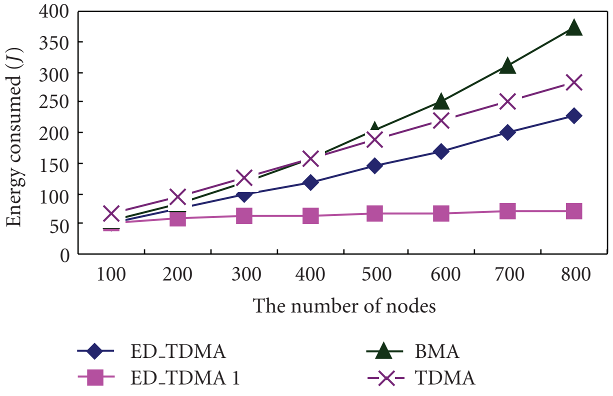

FIGURE 11: Energy consumption versus the number of nodes.

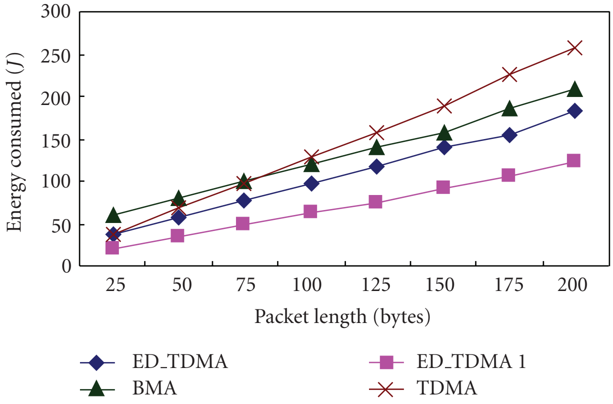

FIGURE 13: Energy consumption versus packet length.

when node density is 0.03 nodes/m^2. As shown in Figure 11, the traditional TDMA wastes more energy due to the idle listening of cluster heads during a round, especially under light traffic load. When p is higher than 0.8, the energy consumed by traditional TDMA is less than ED-TDMA. The reason is that the working time of cluster heads is long but ED-TDMA has more energy consumption in TDMA schedule.

Figure 13 shows the relationship between energy consumption and data packet length after 5000 data cycles. The energy consumption increases with the increasing of packet length. The energy consumed by traditional TDMA is faster than others, which reflects the essence of (9). The more the packet length is, the more energy consumed by traditional TDMA than ED-TDMA.

5.2. Simulation II

5.2.1. Parameters to Impact MAC Protocols. The first goal of MAC protocols designing is energy efficiency. However, less energy consumption does not mean MAC protocols is more energy-efficient. We define energy utility efficiency as network throughput per energy consumption to evaluate MAC protocols, which could be expressed as:

$$\eta_E = \frac{\text{throughput}}{\text{Energy consumed}}. \tag{12}$$

The second goal is scalability. MAC protocols must be scalable with dynamic topology change of WSNs. And the third goal is network efficiency, including latency, throughput and bandwidth utility, and so forth. Obviously, there must be some trade-offs between energy efficiency and network efficiency.

We list some parameters making impacts on MAC protocols in Table 3. N and S could adjust the node density to reflect the scalability of network. P decides transmission probability of sensor node and T_{interval} means packet generation interval. The two parameters control network traffic, which influences the operation of MAC protocols. Packet length L_{data} is another parameter to influence MAC protocols. If L_{data} is small, it causes larger overhead of control information in the packet and decreases energy utility efficiency. We will adjust these parameters in different scenarios to compare contention-based MAC (S-MAC),

Table 3

Parameters	Description
N	Number of sensor nodes
S	Area of monitoring field
P	Transmission probability
T_{interval}	Packet Transmission Interval
L_{data}	Length of Data Packet

cluster-based TDMA (ED-TDMA) and distributed TDMA protocol (LMAC).

5.2.2. Experiment Setup. Simulation parameters are listed in Table 4. To S-MAC, a frame is 1150 ms and its duty-cycle is preset to 10%. LMAC is set to operate with the maximum of 32 slots per frame to ensure that all nodes within a two-hop neighborhood can own a slot. Noticeably, the parameters listed in the table are defined in the scenario of $100 \times 100\,\text{m}^2$ and packet length is 60 bytes. In the following simulations, some parameters will change with different settings.

Table 4: Simulation II parameters.

Protocol	Parameter	Value
S-MAC	Frame length	1150 ms
	Duty-cycle	10%
LMAC	Number of Gateway Node	16
	Frame length	512 ms
	Slot size T_{slot}	16 ms
ED-TDMA	Cluster radius (R)	30 m
	Sensing radius (r)	12 m
	Coverage expectation P_{cover}	95%
	$T_{\text{clustering}}$	500 ms
	T_{tree}	300 ms
	T_{sync}	300 ms
	T_{slot}	16 ms
	T_{collect}	500 ms
	T_{round}	180 s
	T_{rsv}	16 ms
	T_{schedule}	16 ms

5.2.3. Simulation Results

Scenario 1 ($S: 100 \times 100\,m^2$, $L_{\text{data}} = 60\,bytes$, $N: 50 \sim 400$, $P = 1$, $T_{\text{interval}} = 2\,s$). In this scenario, we investigate the influence of the number of nodes, N, and the simulation results are shown in Figure 14 to Figure 16.

Figure 14 plots the average duty-cycle of the three MAC protocols under different node density. Obviously, duty-cycle of S-MAC is a constant value predefined before sensors deployment. To LMAC, nodes are active during their allotted slots in each frame even if they do not have data to send, so the duty-cycle of LMAC keeps constant, too. The duty-cycle of ED-TDMA is higher than the other two when the node density is low. However, when node density is high enough, the duty-cycle of ED-TDMA would be less than S-MAC. The reason is that the active time of nodes of ED-TDMA decreases with the increasing node density because of the intracluster coverage scheme, so that the average duty-cycle of ED-TDMA decreases, too. Similarly, the average energy consumption per node of the three protocols is shown in Figure 15.

Figure 16 shows the energy utility efficiency under different node density. To S-MAC and LMAC, high node density introduces more collisions and lower throughput with the same energy consumption, which decreases the energy utility efficiency. In contrast, the energy utility efficiency of ED-TDMA increases rapidly because its average energy consumption decreases with the increase of node density.

Scenario 2 ($L_{\text{data}} = 60\,bytes$, $N = 500$, $P = 1$, $T_{\text{interval}} = 2\,s$, $S: 50 \times 50\,m^2 \sim 600 \times 600\,m^2$). Figures 17 and 18 show the influence of area of the monitoring field. As shown in Figure 17, the energy consumption of S-MAC and LMAC varies a little under different monitoring area and the energy consumption of ED-TDMA increases. The larger the area is, the more average energy consumption of ED-TDMA. This is because ED-TDMA is a cluster-based protocol, which there would be larger overheads such as cluster management, time synchronization under large monitoring area. For the same reason, the energy utility efficiency of ED-TDMA then decreases drastically with the enlargement of monitoring area while the other two increases as described in Figure 18.

Scenario 3 ($S: 100 \times 100\,m^2$, $L_{\text{data}} = 60\,bytes$, $N = 100$). In this scenario, we study the influence of network traffic. We control network traffic by adjusting packet transmission interval, T_{interval} and transmission probability, P. At first, we set T_{interval} to 2 s and P varies within $[0, 1]$. The results are shown in Figures 19 and 20.

As seen from Figure 20, the average energy consumption of LMAC and ED-TDMA both increases when there are more source nodes. Figure 20 shows that the energy utility efficiency of all the three protocols decreases with the increasing transmission probability. But S-MAC and ED-TDMA decrease more quickly than LMAC.

Secondly, we set P to 0.3 and change T_{interval} from 0.5 s∼ 7 s. Figures 21 and 22 give the results. Smaller T_{interval} means more data packets are generated per slot. As can be seen, the average energy consumption of the three protocols increases when there are more data packets and their energy utility efficiency decrease. But ED-TDMA achieves lower energy consumption and more energy-efficient than the other two.

Scenario 4 ($S: 100 \times 100\,m^2$, $N = 100$, $P = 0.3$, $T_{\text{interval}} = 2\,s$, $L_{\text{data}}: 20 \sim 100\,bytes$). The influence of data packet length is analyzed in this scenario and the results are shown in Figures 23 and 24. As shown in the figures, packet length makes a little impact on S-MAC and ED-TDMA. Because the duty-cycle of LMAC decreases with the increasing packet length, the average energy consumption decreases quickly. Moreover, longer packet length means less control overheads

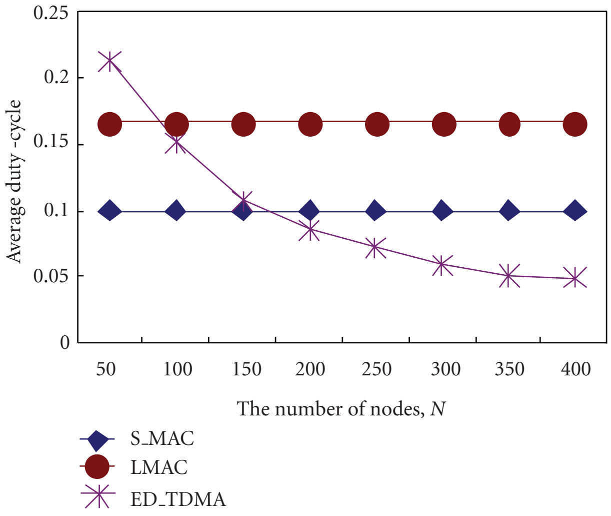

Figure 14: Average duty-cycle versus the number of nodes.

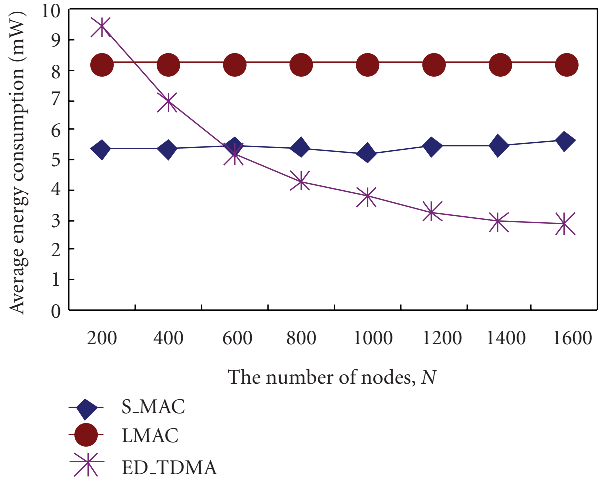

Figure 15: Average energy consumption versus the number of nodes.

in LMAC. So the energy utility efficiency of LMAC increases linearly when packet length increases.

6. Conclusion

In this paper, we presented ED-TDMA, an energy-efficient TDMA protocol for event-driven application for wireless sensor networks. ED-TDMA improves channel utility by changing the length of TDMA frame according to the number of source nodes and saves energy with bitmap-assisted TDMA schedule. In addition, ED-TDMA employs intracluster coverage to prolong network lifetime and to improve system scalability. Compared with contention-based MAC protocol and distributed TDMA scheduling, ED-TDMA performs better for event-driven application in wireless sensor network with high-density deployment and under low traffic.

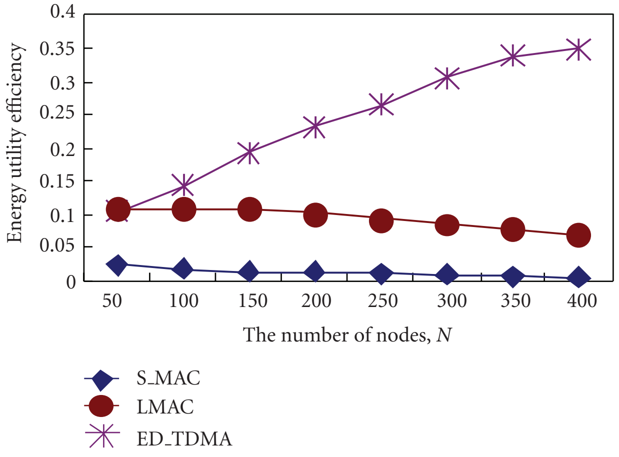

Figure 16: Energy utility efficiency versus the number of nodes.

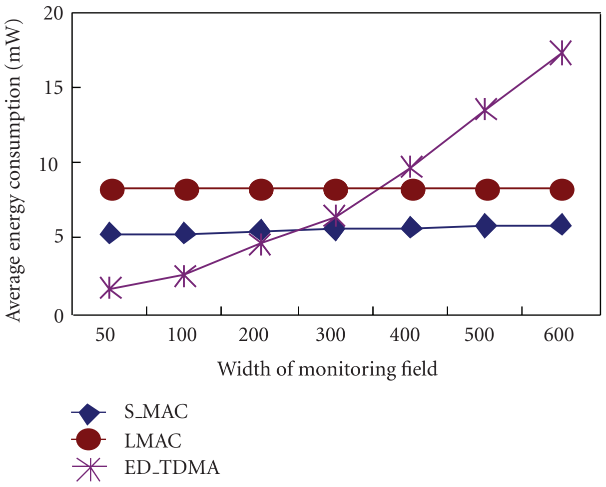

Figure 17: Average energy consumption versus monitoring area.

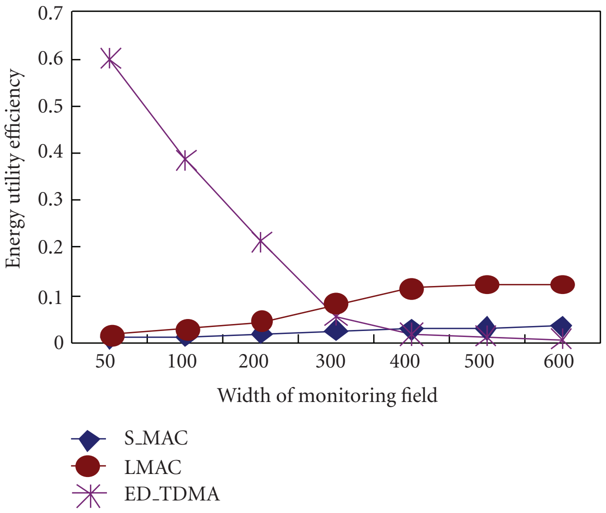

Figure 18: Energy utility efficiency versus monitoring area.

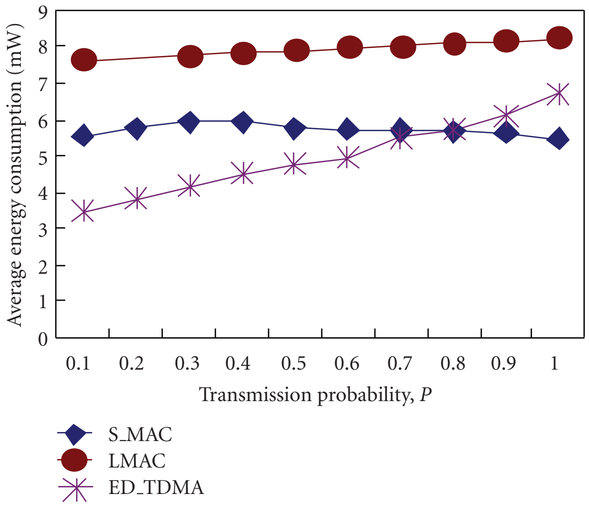

FIGURE 19: Average energy consumption versus transmission probability.

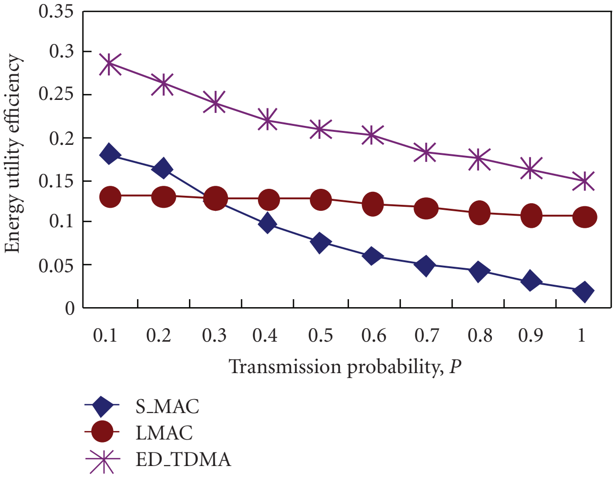

FIGURE 20: Energy utility efficiency versus transmission probability.

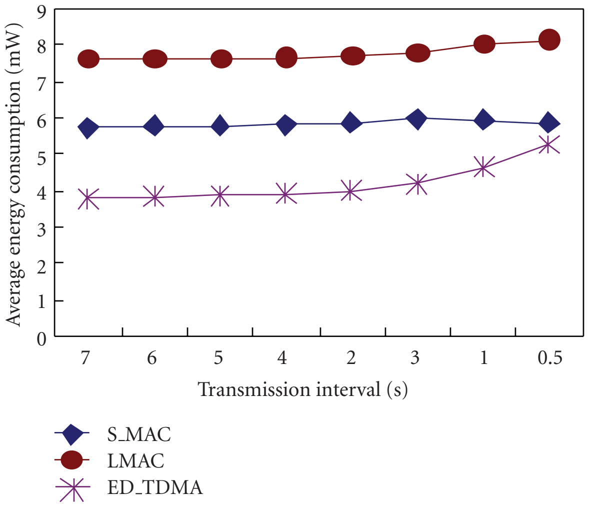

FIGURE 21: Average energy consumption versus transmission interval.

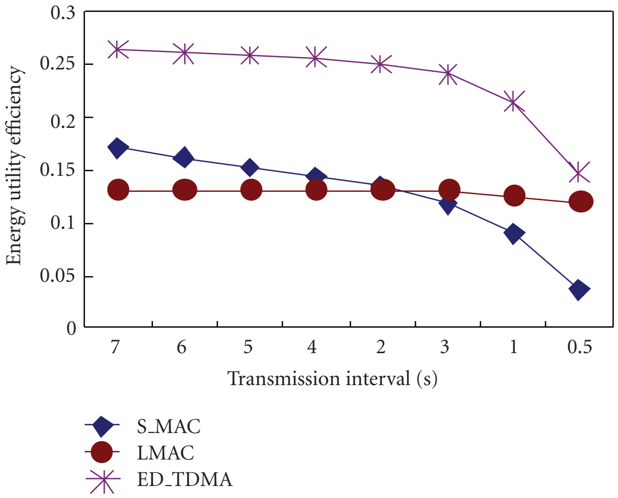

FIGURE 22: Energy utility efficiency versus transmission interval.

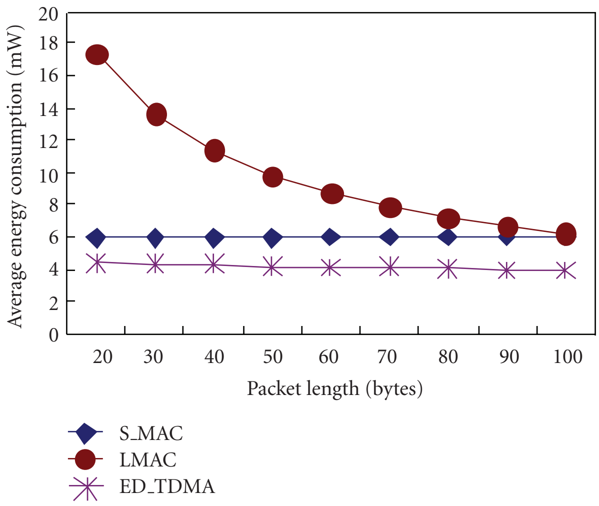

FIGURE 23: Average energy consumption versus packet length.

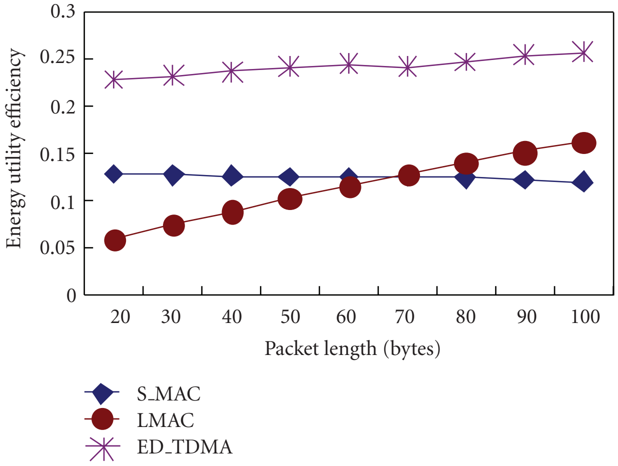

FIGURE 24: Energy utility efficiency versus packet length.

Acknowledgment

This work is partially supported by the National Natural Science Foundation of China under Grant no. 60903158, 60703114, 60903156, and 60873026 and the National Grand Fundamental Research 973 Program of China under Grant no.2006CB303000.

References

[1] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: scalable coordination in sensor networks," in *Proceedings of the International Conference on Mobile Computing and Networking (MobiCOM '99)*, August 1999.

[2] K. Langendoen, *Medium Access Control in Wireless Sensor Networks*, Nova Science, 2008.

[3] LAN-MAN Standards Committee of the IEEE Computer Society, *Wireless LAN Medium Access Control (MAC) and Physical Layer(PHY) Specification*, IEEE, New York, NY, USA, 1997.

[4] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the IEEE INFOCOM*, vol. 3, pp. 1567–1576, New York, NY, USA, June 2002.

[5] T. Van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 171–180, Los Angeles, Calif, USA, November 2003.

[6] S. C. Ergen and P. Varaiya, "PEDAMACS: power efficient and delay aware medium access protocol for sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 7, pp. 920–930, 2006.

[7] K. Arisha et al., "Energy-aware TDMA-based MAC for sensor networks," in *Proceedings of the IEEE Workshop on Integrated Management of Power Aware Communications, Computing and NeTworking (IMPACCT '02)*, New York, NY, USA, May 2002.

[8] S. Kulkarni et al., "TDMA service for sensor networks," in *Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS '04)*, pp. 604–609, Tokyo, Japan, March 2004.

[9] G. Pei and C. Chien, "Low power TDMA in large wireless sensor networks," in *Proceedings of the Military Communications Conference (MILCOM '01)*, vol. 1, pp. 347–351, Vienna, Va, USA, October 2001.

[10] J. Li and G. Y. Lazarou, "A bit-map-assisted energy-efficient MAC scheme for wireless sensor networks," in *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks (IPSN '04)*, pp. 55–60, New York, NY, USA, April 2004.

[11] L. van Hoesel and P. Havinga, "A lightweight medium access protocol (LMAC) for wireless sensor networks," in *Proceedings of the 1st International Workshop on Networked Sensing Systems (INSS '04)*, Tokyo, Japan, June 2004.

[12] H. Gong et al., "Traffic adaptive MAC protocol for wireless sensor networks," in *Proceedings of the International Conference on Computer Networks and Mobile Computing (ICCNMC '05)*, Zhangjiajie, China, August 2005.

[13] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves, "Energy-efficient, collision-free medium access control for wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 181–192, Los Angeles, Calif, USA, November 2003.

[14] I. Rhee, A. Warrier, M. Aia, and J. Min, "Z-MAC: a hybrid MAC for wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, November 2003.

[15] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.

[16] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[17] H. Zhang and J. C. Hou, "Maintaining scheme coverage and connectivity in large sensor networks," in *Proceedings of the NSF International Workshop on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc wireless, and Peer-to-Peer Networks*, 2004.

[18] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration in wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 28–39, ACM Press, November 2003.

[19] Y. Gao, K. Wu, and F. Li, "Analysis on the Redundancy of Wireless Sensor Networks," in *Proceedings of the 2nd ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '03)*, pp. 108–114, San Diego, Calif, USA, September 2003.

[20] M. Liu, H. G. Gong, Y. C. Mao, LI. J. Chen, and LI. Xie, "Distributed energy-efficient data gathering and aggregation protocol for wireless sensor networks," *Journal of Software*, vol. 16, no. 12, pp. 2106–2116, 2005.

[21] RF Monolithics, ASH Transceiver TR1000 Data Sheet, http://www.rfm.com/.