# Self Organization and Energy Efficient TDMA MAC Protocol by Wake Up For Wireless Sensor Networks

Zhihui Chen, *Student Member*
ECE Dept., University of Illinois at Chicago
851 S Morgan St., Chicago, IL, USA
zchen@ece.uic.edu

Ashfaq Khokhar, *Senior Member*
ECE/CS Dept., University of Illinois at Chicago
851 S Morgan St., Chicago, IL, USA
ashfaq@ece.uic.edu

*Abstract* — **This paper proposes novel energy efficient self-organization and medium access control (MAC) protocols for wireless sensor networks. The proposed protocol is based on Time Division Multiple Access (TDMA) principle and are referred to as TDMA. In the proposed protocol a sensor utilizes its assigned slot only when it is sending or receiving information, otherwise its receiver and transmitter are turned off to avoid unnecessary neighbor listening. In order to accelerate receive response, wake-up packets are used to activate a sleeping node. When compared with the existing energy efficient MAC protocols, such as 10% S-MAC, the performance results show that the proposed TDMA-W protocol consumes only 1.5% to 15% of the S-MAC power, yielding 6 to 67 folds enhancement in battery life.**

***Keywords- Wireless Sensor Network; MAC, Self-organization; TDMA***

## I. INTRODUCTION

Wireless sensors networks are gaining popularity across a diverse research community due to their potential usage in pervasive commercial, defense, and scientific applications. Sensor networks are composed of smart sensor nodes interconnected via wireless channels. Each sensor node consists of a sensor coupled with a processor, moderate amount of memory, and transmitter/receiver circuitry. Sensor nodes can sense the environment change and exchange data with its neighbors. Several sensor prototypes such as UC-Berkeley's SmartDust and Mica Motes [11] have been developed. A wide range of potential applications including data collection, environment monitoring, and architecture monitoring are being developed using wireless sensor technology [1],[8].

Sensor networks are different from existing wireless communication networks in following aspects:

- Traffic rate is very low, typical communication frequency is at minutes or hours level.
- Sensor nodes are battery powered and recharging is usually unavailable, so energy is an extremely expensive resource.
- Sensor nodes are generally stationary after their deployment.
- Sensor nodes in the network coordinate with each other to implement a certain function, so traffic is not randomly generated as those in mobile ad hoc networks.

Due to these differences, existing wireless MAC protocols such as IEEE 802.11 [10] are not suitable for sensor networks.

In this paper, we first discuss techniques for self organizing sensor nodes and based on that design an efficient channel access protocol for sensor nodes to transmit/receive data. Since in sensor networks, traffic rate is low and power resource in each node is meager and difficult to replenish, the main objective is to optimize the node lifetime instead of maximizing channel utilization. Several energy-efficient MAC protocols for wireless sensor networks have been proposed [2,3,5-7,9]. One common observation of all these protocols is that the main source of power consumption is *idle listening*. In [4] it is shown that idle listening accounts for more than 90% of the power consumption. In [9] a protocol called S-MAC suggests that nodes follow a listen and sleep cycle, as shown in Figure 1a. Traffic is sent to a destination only during its listening period. This scheme concentrates traffic to certain

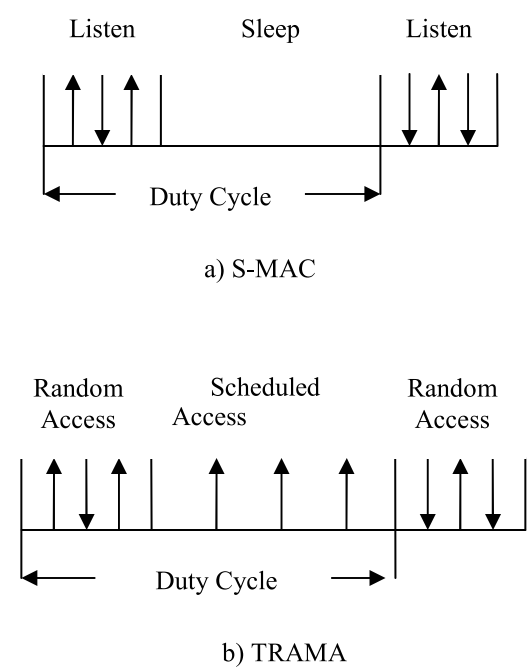



Figure 1: S-MAC [9] and TRAMA [3] MAC protocols

periods and at other times nodes go into a sleep mode by turning off their receivers and thus save energy. To facilitate broadcasting, all nodes start listening at the same time. However, concentrating traffic to fixed periods also increases the contention probability and incurs unnecessary congestion. To resolve contention among simultaneous transmissions, S-MAC proposes to perform the RTS/CTS handshake procedure defined in IEEE802.11 DCF operation. The duty rate or the portion of the listening period of S-MAC should be carefully chosen. If the listening period is long, too much energy would be wasted by idle listening. On the other hand, if the listening period is short, contention probability is high and energy would be wasted by retransmission efforts. In [2] the S-MAC scheme has been improved to adapt duty cycle to the traffic rate.

Another category of protocols is based on scheduling, e.g. TRAMA protocol suggested in [3]. In scheduling based protocols, data transmissions are scheduled in advance to avoid contention. However, in such protocols, besides data transmission, nodes exchange neighbor information periodically to schedule the transmission. In TRAMA, contention-free “scheduled access” and contention-based “random access” are performed alternatively, as shown in Figure 1b. Data transmission is performed in “scheduled access” slot and neighbor information exchange is performed in “random access” slot. The main advantage of TRAMA over S-MAC is improvement in channel utilization. However, the tradeoff is longer delay and higher energy consumption compared with 10% S-MAC [3].

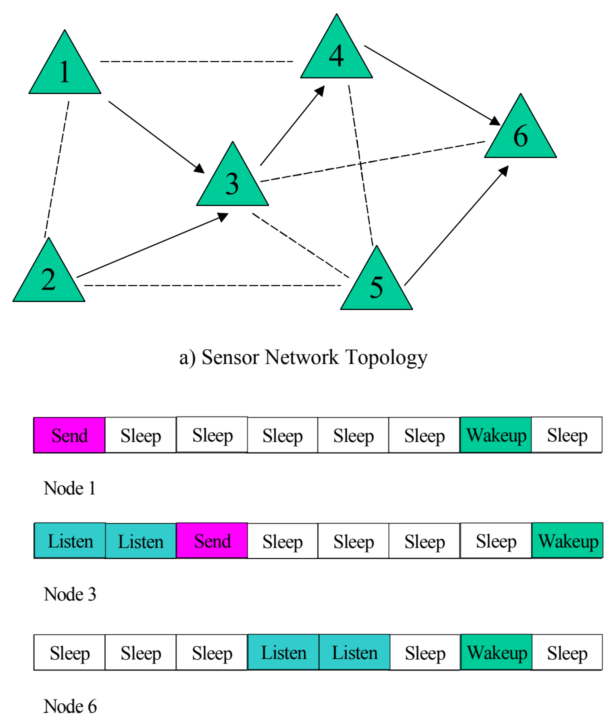
Designing a MAC protocol for sensor networks is analogous to welcoming a guest in a winter season. The guest can only come in when the door is open. If the host always keeps the door open, heating energy will be lost. But if the door is always closed, guests will have to wait for a longer period. The behavior of S-MAC and TRAMA protocols is similar to opening the door periodically and seeing whether there is a guest outside. Another more natural approach is to let the guests knock at the door upon their arrival. This way the host can sense their arrivals and open the door at the right time. Since the sensor activities are usually triggered by events, it is also natural to let traffic transmission triggered by events. As the traffic rate in sensor network is relatively low, we can have fix timings for door knocking for each node. In the following we outline this idea in further detail by employing the Time Division Multiple Access (TDMA) principle. We believe that it is a suitable MAC protocol for sensor networks for its collision-free communication and maintenance simplicity. We refer to the resulting protocol as TDMA-W, which stands for TDMA-wakeup.

In the traditional TDMA protocol in cellular networks, transmission activities of all the nodes *admitted* by the call admission module are organized into TDMA frames. A TDMA frame is divided into time slots and each admitted node is assigned one slot in the frame.

In the case of proposed TDMA-W protocol, all the nodes in the sensor network are considered *admitted.* Each node is assigned two slots in a TDMA frame, the Transmit/Send slot (s-slot) and the Wakeup slot (w-slot). A node always listens the channel during the w-slot and transmits in the s-slot, if needed. To avoid overhearing, only destination nodes need to listen to the transmitter. Other neighbors can turn-off their RF circuits to save energy. In order to activate a node, the source node sends a wakeup packet to the destination node in the w-slot associated with the destination. After receiving the wakeup packet, the destination identifies the source node and starts listening during the s-slot associated with the source node.

For example, consider a sensor network consisting of six nodes, namely, node 1 to node 6, as shown in Figure 2. Let’s assume that the application running on this network is to collect data from all the nodes at node 6. The TDMA-W frame consists of 8 slots, namely, slots 1 to slots 8. Each node is assigned two slots, an s-slot and a w-slot. The s-slot for each node is unique in its two-hop range. In this example, node *i* is assigned slot *i*. Several nodes can share a w-slot. In this example, nodes 1, 4 and 5 are assigned slot 7 as their w-slot and nodes 2, 3 and 6 are assigned slot 8 as their w-slot. The procedure to determine these assignments is part of the self-organization protocol and will be discussed later.

In order to collect data from all the stations, nodes 1 and 2 send their data to node 3, and then node 3 combines these data items with its own data and sends it to node 4. Then node 4 and 5 send their data to node 6. To wake up node 3, node 1 and node 2 need to send a wakeup packet to node 3 in slot 8. If they transmit simultaneously, packets will be corrupted by contention. However, though node 3 cannot decode the source of the wakeup packet, it can search for all of the neighbors, namely, nodes 1, 2, 4 and 5, to find out the incoming traffic. After a while, node 3 finds out that no data is coming from




a) Sensor Network Topology

b) Slot Assignments

Figure 2: An Illustration of TDMA-W protocol

nodes 4 and 5 and then it only listens to nodes 1 and 2. The channel activities of nodes 1, 3 and 6 are illustrated in Figure 2b. For node 1, since there is no incoming data, it only sends data in slot 1 and listens to the wakeup packets in slot 7. For intermediate node 3, it needs to listen to slots 1 and 2 to collect data from nodes 1 and 2, and transmit in slot 3. It also listens to slot 8 for wakeup packets. Node 6, only listens during slots 4, 5 and 7 and does not transmit anything in this example scenario.

The rest of the paper is organized as follows: Section II describes the channel and traffic assumption. The self-organization procedure and proposed medium access protocol TDMA-W is presented in section III. Simulation results are given in section IV and section V concludes the paper.

## II. Channel and Traffic Assumption

Let us consider a simplest wireless channel model where all nodes share a single radio frequency. Both data and control packets are sent and received using this channel. Each node has a fixed communication range. All the nodes within the communication range of a node can receive or decode packets from that node without any error if there is no contention. All the nodes outside the communication range of a node cannot hear or decode any data or control packet from this node. The communication range needs not to be a circle and can be direction dependent. Also, the communication channel is bi-directional and reciprocal, *i.e.,* if node A can hear node B, then node B can also hear node A and vise versa. We assume an ideal physical layer and the only reason for packet loss is transmission contention.

Generally, in sensor networks, the traffic is triggered by events with low possibilities. In order to evaluate the proposed protocol we consider three types of traffic pattern: one-to-all broadcast, all-to-one reduction, and one-hop random traffic. For one-to-all broadcast traffic, the data source broadcasts data to all the nodes in the network. The broadcast traffic may travel several hops to reach the farthest node. For all-to-one reduction, each node sends its information to a data sink. For one hop random traffic, when an event happens, a node transmits a data packet to one of its neighbors. For our experiments we have assumed that events are Poisson distributed.

As described above, the wireless channel is organized as TDMA-W frames. A TDMA-W frame lasts for $T_{frame}$ seconds. This parameter is known to all nodes and is preset before deployment. A TDMA-W frame is divided into slots and each node is assigned one slot for transmission and one slot for wakeup. Networks can be synchronized or non-synchronized. If a network is not synchronized, a node may occupy two slots for transmission or wakeup. Channel activities in a synchronized system always starts at the beginning of a slot, so nodes can achieve synchronization by listening to ongoing traffic. Guard time is kept between two consecutive slots. The guard time is usually much longer than the propagation delay, so time shift can be allowed and no additional synchronization equipments such as GPS are required. For simplification, in the following discussion we assume networks are synchronized.

## III. TDMA-W: Details

### A. Self-Organization

The first step in enabling TDMA-W scheme for sensor networks is to assign time slots to the sensors within each TDMA-W frame. This has to be accomplished in a distributed framework. In the following we present an efficient self-organizing scheme where nodes identify their neighbors and select a proper time slot for transmission and wakeup as part of the protocol.

The IEEE 802.11 based Direct Sequence Spread Spectrum (DSSS) physical layer has a basic rate of 1M bits per second. Let's assume that sensor networks use this rate. Under this assumption, transmission of a 512 byte packet occupies the channel for about 3.9 *msec*. If we assume a TDMA-W frame of 1 second divided into 256 slots, each slot is of 3.9 *msec* duration, thus capable of communicating 512 bytes.

Since transmission rate is low in sensor networks and slot resources are plenty, the probability of two nodes selecting the same slot as their s-slot is very low. We propose that let nodes randomly select their favorite s-slots and then negotiate if two neighboring nodes select the same slot. This problem has resemblance to the graph-coloring problem; however, in our case we cannot assign the same slot to the two hop neighbors either, due to the "hidden terminal" problem. In the following, we outline the proposed self-organization scheme.

1. Each node randomly selects a slot with uniform probability among all slots to be its s-slot (transmit/send slot).
2. During its selected s-slot, each node broadcasts its node ID, its s-slot number, its one-hop neighbors' IDs and their s-slot assignments. It also broadcasts the slot number of any s-slot during which this node has identified a collision.
3. When a node is not transmitting, it turns on its receiver circuit and listens to the traffic from neighbors. The node should record all the information being broadcast by all its neighbors, i.e. their s-slot assignments and their node IDs, and the slot number of any slot being broadcast as a collision-prone slot.
4. If a node determines that it is involved in a collision or finds out that one of its two-hop neighbors has the same s-slot, it then randomly selects an unused slot and go to step 2.
5. If no new nodes are joining in, or s-slot assignments are not changing, or no collisions are detected for a certain period, it implies all neighbor nodes are found and all the s-slots are final.
6. Each node broadcasts the s-slot selections of their two-hop neighbors. Each node identifies an unused slot or any s-slot being used by the nodes beyond its two-hop neighbors and declares it as its w-slot (wakeup slot). Note that w-slots need not be unique.
7. Each node broadcasts its w-slot and the self-organization is complete.

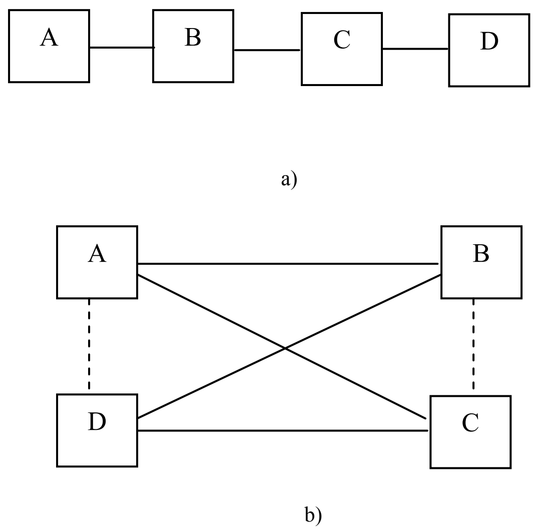



Figure 3: Undetectable one hop and two hop collision

This scheme can detect any two-hop collisions. For example, in Figure 3a, if nodes A and C select the same slot, node B will sense a collision in that slot and report it to A and C. Then nodes A and C can resolve the collision by selecting another slot for transmission.

However, this scheme sometimes cannot detect one-hop collisions. For example, for the same topology of Figure 3a, if nodes B and C select the same slot for transmission, then nodes A and D cannot detect the collision and the network is split into two parts since nodes B and C have no way to find out the existence of each other. To solve this problem, we let a node go to the listening mode in its assigned s-slot with a probability. This gives the node a chance to find out such one-hop collisions. Note that the probability of such a situation is very low. This is because for a sparse network, the number of nodes in two-hop range is small, so the probability of two nodes choosing the same slot is very low. On the other hand, in a dense network though collision probability is high, it is more likely that there exist nodes that can hear nodes B and C, so such a collision can be reported.

Another problem that needs to be taken care of is deadlock. For example, let us consider the situation shown in Figure 3b. Suppose nodes A and D select slot one as their s-slot, while nodes B and C select slot two. Though node B and C can detect the collision between A and D, however, their reports also collide and cannot be received by node A and D. Same scenario happens to node B and C. Then no one can detect the collision. The deadlock problem can be solved by two methods. One method is to listen during transmission slot with a probability, as described in the previous paragraph. Another method is to set a collision counter. If collision in the same slot repeats in the next TDMA-W frame, nodes can realize a deadlock has occurred and switch to another transmission slot.

### *B. The TDMA-W Channel Access Protocol*

After the network is successfully set up, the channel access protocol can be described by the following procedure:

1. Each node maintains a pair of counters (outgoing and incoming) for every neighbor,. These counters are preset to an initial value.
2. If no outgoing data is sent to a node in a TDMA-W frame, the node decrements the corresponding outgoing counter by one; otherwise it resets the counter to the initial value.
3. If no incoming data is received from a neighboring node in a TDMA-W frame, the node decrements the corresponding incoming counter by one. If the counter is less than or equal to zero, the node stop listening to that slot starting from next TDMA-W frame.
4. If an outgoing data transmission request arrives, the node first checks the outgoing counter, if the counter is greater than zero, then the link is considered active and the packet can be sent out during the s-slot. If the counter is less than or equal to zero, a wakeup packet is sent out during the w-slot of the destination node prior to the data transmission.
5. If a node receives a wakeup packet in its w-slot, it turns itself on during the s-slot corresponding to the source node ID contained in the wakeup packet. If a collision is detected in the w-slot, it means more than one node intends to send data. The node then searches all its neighbors for incoming traffic.

The wakeup packet contains only the source and the destination information. The data packet may only contain the destination information and omit source ID since the source ID is determined by the s-slot. If a data packet is to be broadcast to multiple nodes, the destination address contains a special identifier to mark it as a broadcast message. Before sending a broadcast data packet, the node should wakeup all its neighbors that intend to receive this packet. In the case when multiple users share the same w-slot, the destination field of the wakeup message should also be set to a broadcast address.

### *C. Performance Analysis of TDMA-W*

To analyze the delay performance of TDMA-W protocol, let us first fix the position of the w-slot. Suppose the time difference between the w-slot and the s-slot of a node is $T_w$, as shown in Figure 4. The transmission request can arrive at any

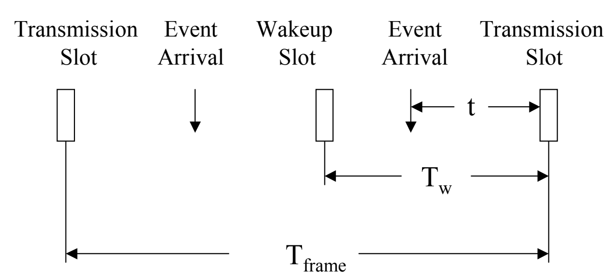



Figure 4: The delay analysis of TDMA-W

time. Suppose the time distance between the transmission request arrival and the s-slot is $t$. If $t > T_w$, or the request arrives before the w-slot, then the destination can be awoke at its s-slot and the delay is $t$. If $t < T_w$, i.e. the event arrives after the wakeup slot, then the data packet has to wait until the wake up slot in the next frame. So delay for this case is $T_{frame}+t$. The average delay for a fixed $T_w$ is:

$$T_D(T_w) = \int_0^{T_w} (T_{frame} + t) p(t) dt + \int_{T_w}^{T_{frame}} t p(t) dt$$

where $p(t)$ is the probability density function. For uniform distribution, $p(t) = 1 / T_{frame}$

$$\text{So } T_D(T_w) = \int_0^{T_w} 1 dt + \int_0^{T_{frame}} \frac{t}{T_{frame}} dt = T_w + \frac{1}{2} T_{frame}$$

Since the w-slots are randomly chosen, so the average delay can be computed as:

$$\overline{T_D} = \int_0^{T_{frame}} T_D(T_w) p(T_w) dT_w = T_{frame}$$

Obviously, the maximum delay for a packet is $2 \times T_{frame}$, if the transmission buffer is initially empty.

## IV. Simulation Results

To verify the performance of the proposed protocol, we have simulated the protocol using MATLAB communication toolbox. For comparison purposes we have also simulated the S-MAC protocol. Nodes are deployed randomly in a 500x500 sq. ft. area. The communication range is 100 feet for all nodes. A sample deployment of 100 nodes is shown in Figure 5. We assume an IEEE 802.11 basic rate of 1M bps as the physical layer transmission rate. The slot length is set to be 4 milliseconds, which is long enough for transmitting a 512-byte packet. $T_{frame}$ is set to be one second, so a TDMA frame has 250 slots.

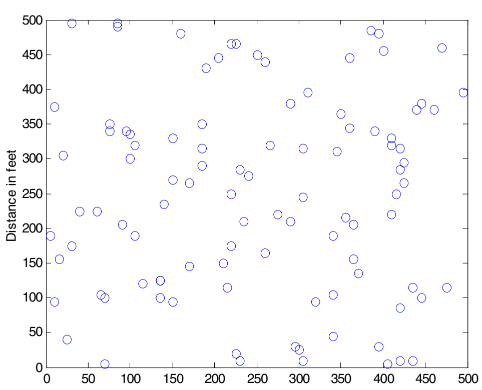



Figure 5: Sample deployment of 100 nodes

| Number of Nodes | Average Time for Self-Organization (Second) | Average Number of One-hop neighbors | Average number of Two-hop neighbors |
|---|---|---|---|
| 50 | 1.628 | 5.12 | 10.84 |
| 100 | 2.408 | 10.41 | 26.13 |
| 200 | 3.626 | 20.87 | 58.47 |

Table 1: Simulation results of self-organization protocol.

First we have simulated the proposed self-organization scheme. We show the results for self-organizing of 50, 100 and 200 nodes in the square area. Each case is simulated by 500 different deployments and results are averaged. The simulation results are shown in Table 1. As expected, time for self-organization increases with the increase in the number of nodes. This is because more collisions happen among neighboring nodes. Since sensor networks are likely to be static and relatively less mobile in nature, the self-organization times are acceptable even for dense networks.

Next we compare the power consumption and delay performance of the proposed TDMA-W protocol. As in [3], we have modeled the power consumption of transmission, receiving/listening and sleeping as 1.83, 1 and 0.001, respectively. We use normalized power consumption of the receiving mode as the basis for comparison. That is, if a node keeps listening to the channel, the average power consumption is one.

As mentioned in Section III, we have simulated three data patterns: one-hop random transmission, One-to-all broadcast, and All-to-one reduction. All data transmissions are event triggered. For One-hop random transmission, event happens at each node independently and a node randomly picks its destination among neighboring nodes. The delay of one-hop transmission is the difference between event arrival time and the time data packet was successfully received. The delay for the broadcast traffic is the difference between the event arrival time and the time when the last node receives the broadcast data packet. For All-to-one reduction, a node first collects the data from the neighboring nodes, concatenates its own data with the incoming data and then forwards the result to the next node. In each slot we only transmit one data packet. The delay for the reduction traffic is the time between the event being triggered and the data sink collecting all the data packets.

In order to reduce power consumption for broadcast and reduction operations, we first construct a spanning tree in the network. The construction of the spanning tree in sensor network is an independent problem and will be addressed in a different work. The data is communicated only along the edge of the spanning tree. Different trees are used for broadcasting and reduction since for broadcasting, we reduce the number of nodes involved in transmission, while in reduction operation every node has to send once except for the data sink, so spanning tree should balance the traffic load and avoid overusing certain nodes.

We simulate the 10% S-MAC and set the active/sleep period to be one second. That is, every node transmits and

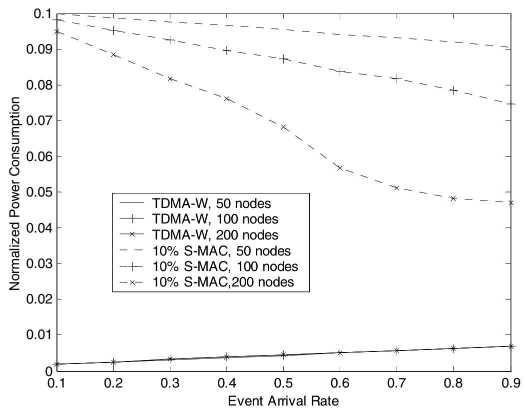



Figure 6: Power consumption of one-hop random traffic.

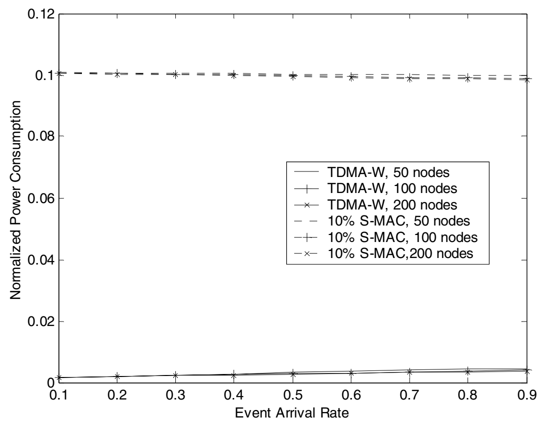



Figure 7: Power consumption of one to all broadcast operation traffic.

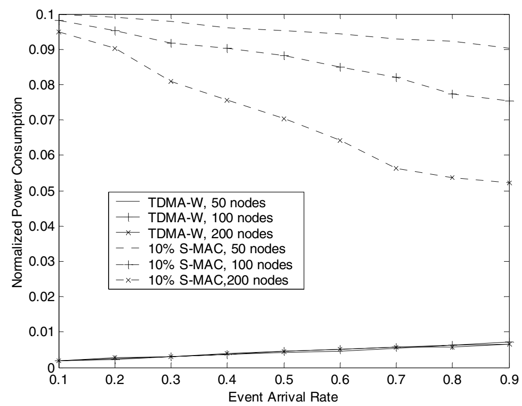



Figure 8: Power consumption of all to one reduction operation traffic.

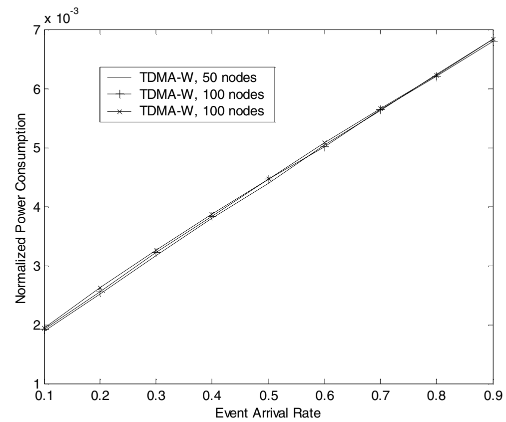



Figure 9: Power Consumption of TDMA-W, one hop traffic

receives traffic for 0.1 seconds and sleeps for 0.9 seconds. The network is synchronized such that all the nodes become active at the same time. The nodes use RTS/CTS frames to reserve channel for node-to-node traffic and use ACK packet to acknowledge the successful transmission. If data or ACK packet is corrupted by collision, the data packet is retransmitted.

All data packets are fixed to be 256 bytes in length and control packets (i.e. RTS, CTS, ACK in S-MAC and Wakeup packet in TDMA-W) are about 20 bytes in length. For simplicity, we assume the energy consumption for transmitting and receiving a control packet is equal to 1/10 of that of a data packet. The initial value for counters is set to 3 and transmission buffer length is set to 50 packets.

Simulation results are shown in Figures 6-10. Both TDMA-W and S-MAC are run for 10 minutes (600 seconds). Figures 6, 7 and 8 show the power consumption of random, broadcast, and reduction traffic, respectively. From these figures we observe that in all the cases, power consumption of the proposed protocol is much lower than 10% S-MAC. For all three types of traffic, the power consumption of TDMA-W ranges from 0.16% to 0.7%. While for 10% S-MAC, energy consumption ranges from 4.7% to 10.1%. So the power consumption of TDMA-W is only 1.5%-15% as much as S-MAC. In other words, the lifetime of TDMA-W networks is 6-67 times longer than 10% S-MAC. Also we observe that the power consumption of TDMA-W increases linearly with the event arrival rate. (Illustrated in Figure 9) This is natural because when event arrival rate increases, more traffic needs to be transmitted and received. However, for random and reduction operations traffic, the average energy consumption of nodes decreases when the event arrival rate and node density increases. This is because when traffic load is low, most of the time no traffic is in the air, so every node keeps listening to the clear channel. While when traffic load is high, there is always traffic in the air and when a node receives an RTS or CTS packet with a destination address of other nodes, it can shut down the receiver and save energy.

Figures 10 and 11 show the average delay of random and reduction operations traffic. For broadcast operation traffic, no channel reservation can be applied and reliable transmission cannot be guaranteed in S-MAC. So not all the nodes can hear the packet, thus broadcasting delay is not available in S-MAC and cannot be compared with TDMA-W.

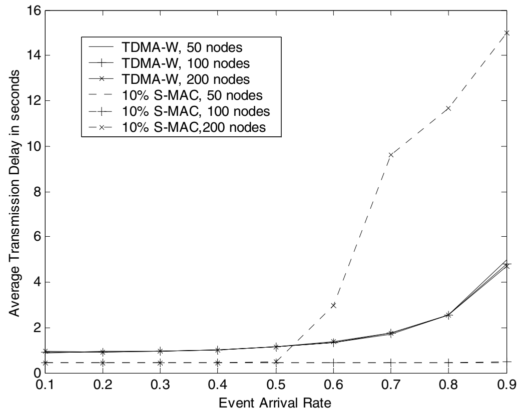



Figure 10: Delay of random tone-hop traffic.

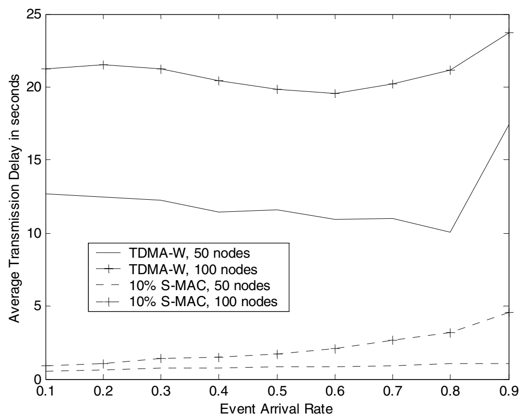



Figure 11: Delay of all to one reduction operation traffic.

Figure 10 shows the average delay of one-hop random traffic. For one-hop random traffic, when traffic load is low (event arrival rate <0.5), the average delay of TDMA-W is around 1 second while the average delay of S-MAC is around 0.45 second. This is because for S-MAC event can arrive at each time. If a transmission request arrives in active period, it can be delivered immediately, if the request arrives in the sleeping period, it has to wait until the active period starts. For TDMA-W, as we have analyzed, the average delay is about one second, which is about twice as much as that of S-MAC. For high traffic load (event arrival rate >0.5), the delay in SMAC increases significantly due to congestion. For 1%-S-MAC, we believe the delay performance results will be closer to TDMA-W. These results will be presented in the full version of the paper.

Figure 11 shows the average delay of reduction operation traffic. We can see the time for TDMA-W to complete a reduction is much higher than S-MAC. This is because traffic can travel multiple hops in a second in S-MAC while in TDMA-W traffic can only travel one hop in each second. So the delay depends on the depth of the spanning tree used for reduction. Note that although reduction traffic suffers long delay, throughput close to 1 reduction/second still can be achieved since the network operates in a pipelined fashion.

## V. CONCLUSION

In this paper, we have proposed efficient protocols for self-organization and channel access control in wireless sensor networks. The protocols, referred to as TDMA-W, employ the well-known TDMA principle. The proposed protocols were verified using extensive simulations. We have shown that the proposed protocol only consumes 1.5% to 15% power of 10% S-MAC. We also show the proposed scheme responds to the event with a delay comparable to S-MAC for one-hop traffic. The proposed protocol is collision free for data traffic so reliable transmission is guaranteed for all types of traffic.

## REFERENCES


[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "A Survey on Sensor Networks", IEEE Communication Magazine, Aug. 2002 p.p. 102-114

[2] T. Dam and K. Langendoen "An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks", International Conference on Embedded Networked Sensor Systems, Nov. 2003, p.p. 171-180

[3] V. Rajendran, K. Obraczka and J.J. Garcia-Luna-Aceves "Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks" International Conference on Embedded Networked Sensor Systems, Nov. 2003, p.p. 181-192

[4] J. Reason and J. M. Rabaey "A Study of Energy Consumption and Reliability in a Multi-Hop Sensor Network" ACM SIGMOBILE Mobile Computing and Communications Review, volume 8 Issue 1, Jan. 2004, p.p. 84-97

[5] A. Safwat, H. Hassanein and H. Mouftah "A MAC-Based Performance Study of Energy-Aware Routing Schemes in Wireless Ad hoc Networks" GLOBECOM'02, Nov. 2002, p.p. 47-51

[6] A. Safwat, H. Hassanein and H. Mouftah "Optimal Cross-Layer Designs for Energy-Efficient Wireless Ad hoc and Sensor Networks" Performance, Computing, and Communications Conference, April 2003, p.p. 123-128

[7] K. Sohrabi, J. Gao, V. Ailawadhi and G. J. Pottie, "Protocols for Self-Organization of a Wireless Sensor Network", IEEE Personal Communications, volume 7, issue 5, Oct. 2000, p.p. 16-27

[8] N. Xu "A Survey of Sensor Network Applications" http://enl.usc.edu/~ningxu/papers/survey.pdf

[9] W. Ye, J. Heidemann and D. Estrin "An Energy-Efficient MAC Protrocol for Wireless Sensor Networks", INFOCOM 2002, June 2002, Vol. 2, p.p. 1567-1576

[10] IEEE802.11 Wireless LAN Medium Access Control (MAC) and Physical (PHY) layer specifications, 1999

[11] http://www-bsac.eecs.berkeley.edu/archive/users/warneke-brett/pubs/hotchips12/tsld001.htm