

Utilization Based Duty Cycle Tuning MAC Protocol for Wireless Sensor Networks

Shih-Hsien Yang*, Hung-Wei Tseng*, Eric Hsiao-Kuang Wu[†], and Gen-Huey Chen*

* Dept. of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, R.O.C.

[†]Dept. of Computer Science and Information Engineering, National Central University, Chung-Li, Taiwan, R.O.C.

Abstract—In this paper, we propose U-MAC, a medium access control protocol designed for wireless sensor networks. Nowadays, wireless sensor network are formed by a great quantity of sensor nodes, which are generally battery-powered and may not recharge easily. Consequently, how to prolong the lifetime of the nodes is an important issue while designing a MAC protocol. However, lowering the energy consumption may result in higher latency. Addressing on such tradeoff, U-MAC balances the tradeoff by utilization based tuning of duty cycle and selective sleeping after transmission. The experiment results show that our proposed U-MAC saves energy about 43% and reduce latency by 65% from S-MAC in a chain topology. In the cross topology, U-MAC also achieves 32% energy saving and 45% latency reduction from S-MAC.

I. INTRODUCTION

Being an emerging technology, wireless sensor networking has various applications that can help to sense and track environmental attributes. Sensor network is mainly composed of a large number of sensor nodes. The sensor nodes are often battery-powered and equipped with an on-board processor carrying out simple computation. Therefore the sensor nodes can only send out the required data to the data gathering sink.

Because of the limited power supply on sensor nodes, energy consumption is often an important design consideration. Ye et al. proposed S(Sensor)-MAC [1] which is specifically designed to reduce energy wastage on IEEE 802.11 based sensor nodes.

The improvements of S-MAC are as following:

- 1) Periodically listen and sleep schedule: The traffic loads of many sensor applications are not very heavy, that is, sensor nodes spend lots of time in idle listening state, which consumes as much power as receiving data. S-MAC propose sensor nodes should periodically listen and sleep to solve idle listening problem. A complete listen and sleep cycle is called a frame. The portion of listening time in a frame is called duty cycle. Nodes will broadcast their schedule to their neighbors, and nodes adopt the same listen and sleep schedule will form a virtual cluster.
- 2) Overhearing avoidance: In traditional IEEE 802.11, nodes listen to all transmissions from their neighbors, even when the packets are not destined to them. To avoid such overhearing problem, S-MAC proposes that nodes hearing RTS or CTS packets not directed to them should go to sleep until the data transmission ends.

- 3) Message passing: A long message in 802.11 networks will be divided into several fragments, and there will be an RTS/CTS/DATA/ACK handshaking process for each fragment of that message, which results in control message overhead and higher retransmission penalty. S-MAC solves this problem by transmitting fragments belonging to the same message with only one handshaking.

However, S-MAC still has its own problems. For example, the significantly increased latency due to sleep delay. Besides, the uniform assignment of duty cycle of all nodes across a whole network also reduce the energy efficiency of S-MAC.

Addressing on the problems of S-MAC, we propose U-MAC. U-MAC provides three main modifications over S-MAC, the various duty cycle scheme to assign different duty cycles for different tasked nodes, the utilization based approach for calculating suitable duty cycles and selective sleeping after transmission to further reduce energy wastage in S-MAC.

Similar to U-MAC, DSMAC [2] also proposes various duty cycles for nodes. Distinct from the subtle tuning in U-MAC, DSMAC tunes duty cycle by doubling or halving according to delay and the emptiness of packet queue. We setup experiments using ns-2. The experiment results show that U-MAC effectively reduce energy consumption by 43% on average and latency by 65% from S-MAC in a chain topology. In the cross topology, U-MAC also achieves 32% energy saving and 45% latency reduction from S-MAC. Comparing with DSMAC, U-MAC always takes advantages from the tuning mechanism used.

The rest of this paper is organized as follows. Section II discusses the related works. Section III describes our proposed scheme. Section IV presents the simulation results to show the energy efficiency of our proposed scheme. Section V concludes this paper.

II. RELATED WORK

Due to the energy constraint of sensor nodes, energy saving is a key subject to sensor network MAC design. Current TDMA-based protocols are generally energy conserving, but they usually requires the nodes to form clusters like Bluetooth [3] and LEACH [4]. However, it is difficult for the clusters to dynamically change its frame length and time slot assignment when a member joins or leaves the cluster. Therefore the scalability and mobility of the TDMA-based protocols isn't as good as contention-based protocols.

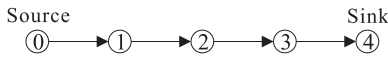


Fig. 1. The topology used in evaluating the energy consumption of S-MAC

In addition to the TDMA-based protocols, out-band signaling is another way to save energy. PAMAS [5] and STEM [6] use another channel to separate the control signal, so the data channel can turn off while it is not transmitting or receiving and then reduce energy consumption. Nevertheless, it requires a second channel to handle the control signals, which may not be suitable for the current sensor nodes due to the cost of them.

The well-known wireless network protocol IEEE 802.11 is in-band signaling. S-MAC [1] improves the energy consumption of IEEE 802.11 to satisfy the requirement of sensor networks. To better improve the energy efficiency based on S-MAC, several researchers propose their solutions [2] [7] [8] [9] [10]. DMAC [7] proposed to deliver data along the data gathering tree and the data flow is unidirectional. The schedule is staggered, so nodes on the path wake up sequentially to forward a packet to next hop. But the disadvantage of using the tree structure is that nodes cannot communicate to each other arbitrarily. The scheme in [8] transmits the queued message at start of the frame in bursts of variable lengths, and then listens for a timeout period. If the node hears nothing during the timeout period, it goes to sleep until the next frame. This scheme saves energy, however, it may also increase the sleep delay. [2] proposed to halve or double the duty cycle based on the measurement of the energy consumption level and delay, therefore it can adjust the schedule to adapt different traffic loads.

III. U-MAC

S-MAC improves the energy consumption of traditional IEEE 802.11, however, several problems arise from the adoption of S-MAC.

- 1) Uniform duty cycle: In a typical sensor network, nodes may take different tasks such as data source nodes, intermediate nodes, and sink nodes. Furthermore, different types of nodes also have different transmission behaviors and traffic loads. In S-MAC, a fixed duty cycle is assigned across the whole network. The uniform duty cycle may result in energy wastage on nodes with low traffic load, and long queueing delay on nodes with heavy traffic. In the chain topology shown in Figure 1, we find that the source node 0 achieves best energy saving with 20% duty cycle, but the intermediate node 2 with 35% duty cycle. The observation reflects the problematic uniform duty cycle in S-MAC.
- 2) Latency: Some sensor applications, such as environment surveillance, require sensing data being sent from source to sink within a certain period. Under some energy consideration, we tend to set sensor nodes in low duty cycle. In S-MAC, a node cannot forward a packet until the packet's receiver wakes up, and such latency

is called sleep delay. As described in [1], the sleep delay is about half of a complete cycle of listening and sleeping on average. To fulfill the delay constraint of such applications, the sensor MAC protocol must be designed under the consideration of latency.

- 3) Idle listening after transmission: As described in Section I, S-MAC proposes periodically listen and sleep scheme to reduce idle listening. However, a node will keep idle listening until next frame after data transmission if the transmission ends inside the sleep period in schedule.

Addressing on the above problems of S-MAC, we propose our MAC protocol based on S-MAC, and our proposed protocol is described in the following paragraphs.

A. Various Duty Cycle

As stated above, the uniform assigning duty cycle in S-MAC results in energy wastage under some circumstances. Addressing on the problem, our proposed scheme does not assign the same duty cycle for nodes, and each node can be assigned different periodically listen and sleep schedules with different duty cycle. As in S-MAC, we force sensor nodes to exchange schedules and synchronize clock with their neighbors with a fixed period. However, nodes receiving SYNC packets will not adopt the same schedule from their neighbors as in S-MAC. To avoid unnecessary retries of RTS due to missing updated schedules from neighbors, we also piggy-back the time of its next sleep in ACK packets.

In order to satisfy the requirement of each nodes, we also proposed a method for optimizing the duty cycles on each nodes, and it is described in the next section.

B. Utilization Based Tuning of Duty Cycle

Every nodes in a sensor network have their own different traffic loads according to what tasks they take and where the locations they are. To reflect the different traffic loads, we adopt the utilization of each node as the evaluation metric. Upon synchronization time, the node calculates its utilization after the last synchronization time, and adjusts its duty cycle according to the calculated utilization, and then send the new schedule to its neighbors via broadcasting. The detailed algorithm is summarized as Algorithm 1.

Algorithm 1 sendSYNC

- 1: $U \leftarrow \frac{T_{rx} + T_{tx}}{T_{rx} + T_{tx} + T_{idle}}$
 - 2: $\bar{D} \leftarrow \frac{D}{packet\ count}$
 - 3: **if** $U > U_{high}$ and $duty\ cycle < DC_{max}$ **then**
 - 4: $duty\ cycle \leftarrow duty\ cycle + n\%$
 - 5: **else if** $U < U_{low}$ and $duty\ cycle > DC_{min}$ and $\bar{D} < D_{max}$ **then**
 - 6: $duty\ cycle \leftarrow duty\ cycle - n\%$
 - 7: **end if**
 - 8: $D \leftarrow 0$
 - 9: $packet\ count \leftarrow 0$
-

As shown in Algorithm 1, the node first calculates the utilization U as $\frac{T_{rx} + T_{tx}}{T_{rx} + T_{tx} + T_{idle}}$, where the T_{rx} stands for total

Algorithm 2 handleDATA

- 1: $d \leftarrow$ the value of delay field in data packet
 - 2: $D \leftarrow D + d$
 - 3: $packet\ count \leftarrow packet\ count + 1$
-

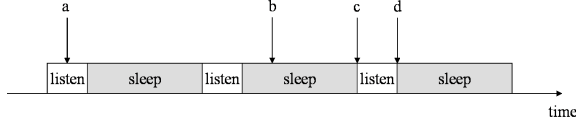


Fig. 2. A transmission may ends at scheduled sleep time or listen time

receiving time, T_{tx} stands for total transmitting time, and T_{idle} stands for total idle listening time during this synchronization interval. Both of the total receiving and total transmitting time are the time that a node staying in receiving and transmitting states respectively. The total idle listening time is the time that a node staying in listening state without involving in any transmission process. If the utilization U of a node is high, the traffic load is too heavy for the current listening schedule to afford. Therefore, we should increase the duty cycle to fit the requirement of such heavy traffic load. In our proposed scheme, we define U_{high} as the threshold of high traffic load. Once upon the node is considered as high utilization, the duty cycle will increased by $n\%$. Larger n results in faster adaption of traffic loads, but smaller n avoids the excessive response and helps finding out best schedule under different traffic loads. Besides, to provide the guarantee of maximum energy consumption, our proposed algorithm also sets the upper limit of duty cycle, DC_{max} . A node stops increasing its duty cycle if the current duty cycle reaches DC_{max} . Similarly, a node will decrease its duty cycle as the utilization U is smaller than U_{low} , the threshold of low traffic, and stop decreasing its duty cycle if reach the defined lower bound of duty cycle, DC_{min} .

Concerning the latency, a node will also stop decreasing its duty cycle if the average sleep delay of its packet senders, \bar{D} , is more than the tolerable sleep delay, D_{max} . To calculate \bar{D} , the sending node must compute the suffered sleep delay (d) during the current transmission, and attach d in the following data packet. The sleep delay (d) is defined as the delay between receiving the packet from upper layer and sending out RTS. The receiving node extracts d from data packet and adds d to the total sleep delay D , and refresh the $packet\ count$, which presents the number of packets received during this synchronization interval. Algorithm 2 summarizes the processing of data packets on receiving nodes. With the accumulated sleep delay D , the node can calculate the average sleep delay \bar{D} by $\frac{D}{packet\ count}$.

C. Selective Sleeping After Transmission

In S-MAC, a node goes to sleep due to scheduled sleep time arrived or after reception of control packets such as RTS and CTS not destined to itself. However, the MAC protocol can get extra energy saving from sleeping after transmission.

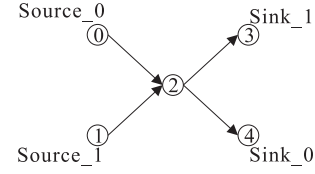


Fig. 3. The cross topology used in experiments

As shown in Figure 2, a transmission may end at scheduled listen time like "a", or scheduled sleep time like "b". If a transmission ends at scheduled sleep time b , the node will keep listening until the next scheduled listen time c , so that between b and the next scheduled listen time c , the energy is wasted.

To avoid the above energy wastage, we propose that a node should go to sleep "selectively". When transmission is finished, a node checks if it is at scheduled sleep time, and goes to sleep if it's at scheduled sleep time. The sleep will not result in additional delay since the node is considered as sleep on its neighboring nodes, and no node will send packet for the node.

Assuming packet transmission will end with equal probability within a frame at the receiver, let T_{frame} denote the complete cycle time of listen and sleep schedule. The average sleep time due to selective sleeping is

$$\frac{T_{frame}}{2}$$

similar to the calculation of sleep delay. As described in S-MAC, the T_{frame} is

$$T_{frame} = \frac{T_{listen}}{duty\ cycle}$$

and T_{listen} is a fixed length listen period within a frame. Therefore, we can expect that selective sleeping can achieve energy saving per transmission as:

$$\frac{T_{listen}}{2 \times duty\ cycle} \times (P_{idle} - P_{sleep})$$

where P_{idle} and P_{sleep} stand for power consumption of idle and sleep state, respectively. According to the above formula, we can expect that the energy saving from selective sleeping can be more significant when we tuned down the duty cycle.

IV. SIMULATION AND RESULTS

In this section, we will describe the way how we evaluate our proposed U-MAC, and the experimental results which demonstrate both energy consumption and latency. The results include the comparison among U-MAC, DSMAC, and S-MAC.

A. Experiment Setup

The simulations in our experiment are done on ns-2(ns-2.27) network simulator. We compared U-MAC with DSMAC because its scheme is similar to U-MAC in various duty cycle. However, the duty cycle tuning mechanism in DSMAC only

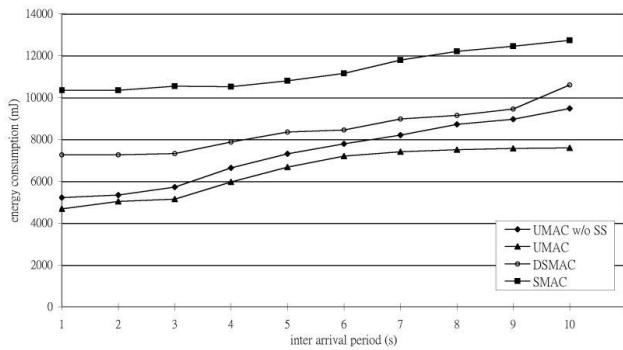


Fig. 4. The energy consumption of U-MAC without sleep after transmission (U-MAC w/o SS), U-MAC with selective sleeping after transmission (U-MAC), DSMAC, and S-MAC in the topology of Figure 1

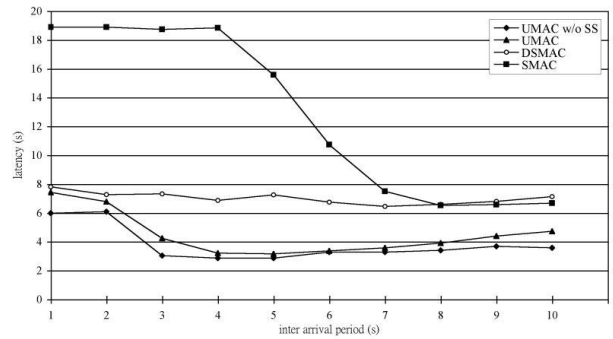


Fig. 5. The latency of U-MAC without sleep after transmission (U-MAC w/o SS), U-MAC with selective sleeping after transmission (U-MAC), DSMAC, and S-MAC in the topology of Figure 1

depends on packet queue and delay. DSMAC also reserves adopting schedule from neighboring nodes, which is excluded from U-MAC. To show how much we can benefit from proposed utilization based duty cycle tuning mechanism, we also setup experiments for U-MAC without selective sleeping after transmission. We include S-MAC since U-MAC is mainly based on the concept of S-MAC.

The topology used to evaluate our proposed scheme is shown in Figure 1, where node 0 is the data source node and packets are forwarded to data sink node 4 by node 1 through node 3.

In our experiments, we set the transmitting power to 24.75 mW, receiving power and idle listening power to 13.5 mW, and sleeping power to 15 μ W. The bandwidth is set to be 20 kbps. Each data packet size is 400 Bytes, and 10 Bytes for control packets including RTS, CTS, and SYNC. The maximum duty cycle threshold DC_{max} and minimum duty cycle threshold DC_{min} are set to 40% and 10% respectively. Besides, U_{high} is 0.3 and U_{low} is 0.15. The increment and decrement amount of duty cycle, n , are set to 2%. All nodes are initialized as 20% duty cycle. In the aspect of performance guarantee, we also set D_{max} to 2 seconds, which is the same as in DSMAC.

The duty cycle is set to 10% in S-MAC, and the parameters for DSMAC are set to be the same as those in [2]. To show the performance of U-MAC, we evaluate total energy consumption of the whole topology and end-to-end latency under different packet inter-arrival periods varying from 1 second to 10 seconds.

We also test our proposed protocol with another topology shown in Figure 3. In the topology, node 0 and node 1 generate data to sink node 4 and node 3, respectively. The only intermediate node is node 2, and we can expect that the traffic load of node 2 is very heavy. The parameters used in testing the topology is the same as in the chain-hop topology.

B. Experiment Result

[t]

The simulation results of the chain topology are shown in Figure 4 and Figure 5. Our proposed U-MAC scheme without selective sleeping after transmission is denote as "U-MAC w/o

SS" in the result figures. The complete U-MAC including all parts of proposed scheme is denote as "U-MAC". The DSMAC and S-MAC protocol are identified as "DSMAC" and "S-MAC".

Figure 4 shows the total energy consumption of all nodes in Figure 1 for all the protocols we tested. As shown in the figures, U-MAC saves significant energy from original S-MAC even selective sleeping is not adopted, especially when traffic load is high. U-MAC w/o SS achieves about 49% energy saving when inter-arrival period is 1 sec, and the network is quite saturated under such inter-arrival period. When inter arrival period is set to 10 secs, U-MAC w/o SS still saves energy from S-MAC by 26%. The average energy saving achieved by U-MAC w/o SS is about 35% of S-MAC. Comparing the result with DSMAC, U-MAC w/o SS always saves more energy than DSMAC, and this points out that U-MAC provides a better solution for dynamic duty cycle tuning than DSMAC. When selective sleeping after transmission is adopted, the energy consumption is further reduced. The U-MAC with SS achieves about 43% energy saving from S-MAC on average. As the curve shown in Figure 4 also reflects that the effect of selective sleep is more significant when traffic load is lower. Because U-MAC tends to decrease the duty cycle when utilization, which is related to traffic load, is lower, this results in more significant energy saving from selective sleeping as we explained in Section III-C.

Figure 5 shows the average end-to-end latency per packet in our experiments. As shown in the figure, U-MAC significantly reduces latencies through all experiments. U-MAC w/o SS reduce latency by 65% on average, and reduce latency by 85% from S-MAC when inter-arrival period is set to 4 secs. DSMAC can only improve latency when traffic load is high, the latencies cannot benefit from DSMAC when traffic load is light, like the cases with inter-arrival periods more than 8 secs. Even when traffic load is light, our proposed U-MAC still reduce latency effectively. For example, in the test with 10 secs inter-arrival period, U-MAC w/o SS reduces latency by 46%. The selective sleeping after transmission may slightly raise the calculated utilization upon tuning duty cycles, since the *total idle listening time* is reduced by this scheme. However,

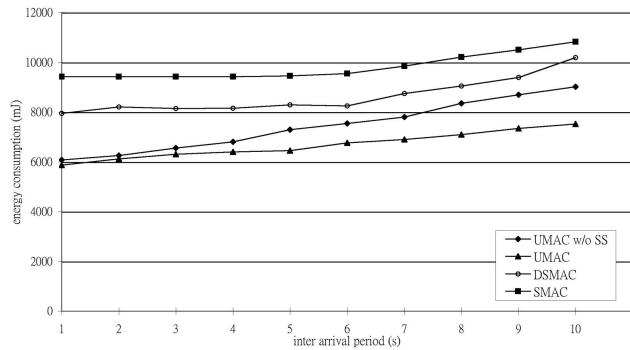


Fig. 6. The energy consumption of U-MAC without sleep after transmission (U-MAC w/o SS), U-MAC with selective sleeping after transmission (U-MAC), DSMAC, and S-MAC in the topology of Figure 3

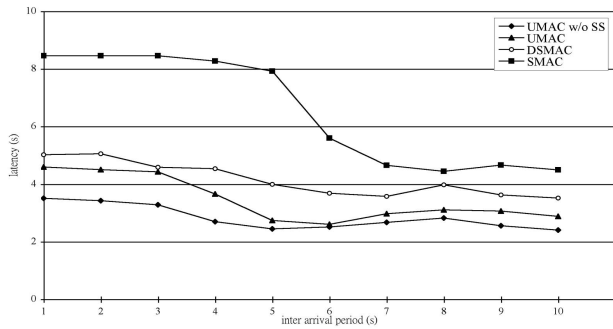


Fig. 7. The latency of U-MAC without sleep after transmission (U-MAC w/o SS), U-MAC with selective sleeping after transmission (U-MAC), DSMAC, and S-MAC in the topology of Figure 3

as shown in the result of Figure 5, this does not make great impacts on latencies, and the latencies of U-MAC with SS is reduced to 40% of S-MAC on average.

In the cross topology experiment, where the traffic load is concentrated on node 2, U-MAC still shows better energy efficiency than S-MAC and DSMAC. As shown in Figure 6 and Figure 7, U-MAC saves about 32% energy and 45% latency from S-MAC.

Figure 8 shows the duty cycles of U-MAC during the experiment of 10 sec inter-arrival period. The traffic loads of intermediate nodes are relatively high, and U-MAC successfully reflects this fact on the duty cycles of nodes 1, 2, and 3. Since data source node 0 only reports the sensing result and seldom forwards packets, lowering duty cycle does not hurt latency and benefit energy saving, and U-MAC assigns very low duty cycle for node 0. The traffic load of data sink node 4 is not as heavy as intermediate nodes since node 4 only receives data. As shown in the figure, U-MAC also tunes the duty cycle of node 4 to be lower than intermediate nodes.

V. CONCLUSION AND FUTURE WORK

There are various kinds of sensor network applications. They may require lower energy consumption, or less latency, or both of them. We propose U-MAC to find the balance between energy consumption and latency for different sensor network applications in this paper. Several schemes are used,

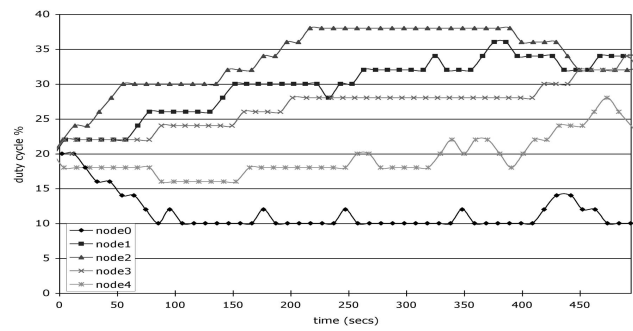


Fig. 8. The duty cycle for all nodes in Figure 1. The inter-arrival period is set to 10 secs

which include various duty cycle, utilization based tuning of duty cycles, and selective sleeping after transmission. We let the sensor nodes adjust their duty cycle automatically, aiming at lowering both energy and latency. We also simulate our schemes with a chain topology, and the simulation results show that the duty cycle tuning method assigns proper duty cycles for nodes with different tasks and network locations. By our proposed U-MAC, not only the total network energy consumption, but the end-to-end latency is significantly improved.

As mentioned in Section II, TDMA based protocols consume lower energy since TDMA protocols avoid the overhead of contending the channel. In the future, we will try to explore the hybrid approach which combines contention-based protocols like S-MAC and our proposed U-MAC with the concept of contention free MAC protocols.

REFERENCES

- [1] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of IEEE INFOCOM*, June 2002, pp. 1567–1576.
- [2] P. Lin, C. Qiao, and X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," in *WCNC 2004 - IEEE Wireless Communications and Networking Conference*, vol. 5, no. 1, March 2004, pp. 1522–1527.
- [3] J. C. Haartsen, "The bluetooth radio system," *IEEE Personal Communications Magazine*, pp. 28–36, 2000.
- [4] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocols for wireless microsensor networks," in *Proceedings of the Hawaii International Conference on Systems Sciences*, January 2000.
- [5] S. Singh and C. Raghavendra, "PAMAS: Power aware multi-access protocol with signalling for ad hoc networks," *ACM Computer Communication Review*, vol. 28, no. 3, pp. 5–26, July 1998.
- [6] C. Schurgers, V. Tsitsis, and M. Srivastava, "Stem: Topology management for energy efficient sensor networks," in *Proceedings of the 2002 IEEE Aerospace Conference*, March 2002.
- [7] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "An adaptive energy-efficient and low-latency MAC for data gathering in sensor networks," in *Proceedings of 18th International Parallel and Distributed Processing Symposium*, April 2004, pp. 224–231.
- [8] T. van Dam and K. Langendoen, "An adaptive energy efficient MAC protocol for wireless sensor networks," in *Proceedings of the First ACM Conference on Embedded Networked Sensor Systems*, November 2003, pp. 171–180.
- [9] 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*, 2004, pp. 95–107.
- [10] H. Pham and S. Jha, "An adaptive mobility-aware MAC protocol for sensor networks (MS-MAC)," in *Proceedings of the 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, October 2004.