Improving IEEE 802.11 Power Saving Mechanism

Eun-Sun Jung[1] and Nitin H. Vaidya[2]

[1] Dept. of Computer Science, Texas A&M University, College Station, TX 77843, USA
Email: esjung@cs.tamu.edu
[2] Dept. of Electrical and Computer Engineering, and Coordinated Science Laboratory,
University of Illinois, Urbana, IL 61801 USA, Email: nhv@crhc.uiuc.edu

Technical Report

July 7, 2004

***Abstract*— This paper presents a mechanism that improves the energy efficiency of PSM (Power Saving Mechanism) specified for DCF (Distributed Coordination Function) in IEEE 802.11. In PSM, time is divided into beacon intervals. At the start of each beacon interval, every node stays awake for a duration, called the ATIM Window. During the ATIM window, nodes exchange control packets to determine whether they need to stay awake for the rest of the beacon interval. Past research results show that the size of the ATIM window has a significant impact on energy savings and throughput achieved by nodes. This paper proposes a simple mechanism that adjusts ATIM window size to improve the energy conservation of PSM. We also allow nodes to enter the doze state in the middle of beacon interval if they have completed all the transmissions that are explicitly announced during the ATIM window. Simulation results show that the proposed solution improves performance and energy savings over PSM in IEEE 802.11.**

***Index Terms*— IEEE 802.11, MAC, energy efficiency, power saving mode, wireless network, dynamic ATIM window.**

I. Introduction

Over the past few decades, a large volume of research has been conducted in the area of wireless networks. In contrast to wired network, energy is a critical resource in the design of wireless network since wireless devices are often powered by batteries. Many techniques have been proposed to increase battery lifetime. In this paper, we propose a power saving mechanism, which improves the one specified for DCF (Distributed Coordination Function) in an IBSS(Independent BSS) of IEEE 802.11.

IEEE 802.11 consists of two components: PCF and DCF. PCF (Point Coordination Function) is a centralized medium access control protocol, whereas DCF is a fully distributed protocol [9]. IEEE 802.11 specifies a power saving mechanism for both PCF and DCF. This paper focuses on the power saving mechanism for DCF in a wireless LAN environment wherein all nodes are in each other's transmission range.

Fig. 1 illustrates the power saving mechanism (hereafter referred as PSM) in DCF. Time is divided into beacon intervals [9]. In this paper, we do not consider the problem of time synchronization and assume that beacon intervals begin and end approximately at the same time at all nodes. At the start of each beacon interval, each node must stay awake for a fixed time interval, called the *ATIM (Ad-hoc Traffic Indication Message) window*.

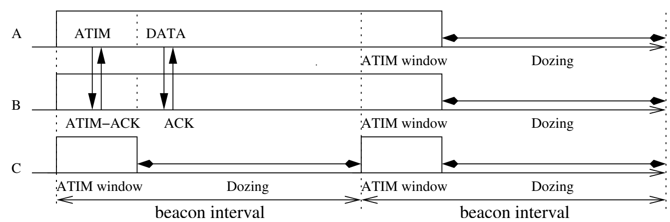


Fig. 1. PSM: Node A transmits an ATIM frame to B during the ATIM window to announce a buffered packet. Node B replies by sending an ATIM-ACK, and both A and B stay awake during the entire beacon interval. After the ATIM window finishes, A transmits the actual data to B. Since node C does not have any packet to send or receive, it dozes after the ATIM window.

In PSM, the source node buffers packets for the destination node that is in the doze state, and these buffered packets are announced during a subsequent ATIM window. For instance, in Fig. 1, node A announces a packet destined for node B by transmitting an "ATIM frame" during the ATIM window. The transmission of an ATIM frame is performed using the CSMA/CA (collision avoidance) mechanism specified in IEEE 802.11. When a node has sent an ATIM frame to another node, such as node A in our example, it remains awake for the entire beacon interval. A node that receives an ATIM frame replies by sending an ATIM-ACK. Such a node remains awake for the entire beacon interval after transmitting the ATIM-ACK. In our example, node B sends an ATIM-ACK to node A and remains awake for the rest of the beacon interval. Transmission of one or more data packets from node A to node B can now take place during the beacon interval, after the end of the ATIM window. A node that has no outstanding packets to be transmitted can go into the doze state at the end of the ATIM window if it does not receive an ATIM frame during the ATIM window. In Fig. 1, node C dozes after the ATIM window, thus saving energy. All dozing nodes again wake up in PSM at the start of the next beacon interval.

This research is supported in part by National Science Foundation grant 01-25859. The report is derived from a manuscript date January 20, 2004.

In PSM, all nodes use the same (fixed) ATIM window size as well as identical beacon intervals [9]. Since the ATIM window size critically affects throughput and energy consumption, a fixed ATIM window does not perform well in all situations, as shown in [11]. If the ATIM window is chosen to be too small, there may not be enough time available to announce buffered packets (by transmitting ATIM frames), potentially degrading throughput. If the ATIM window is too large, there would be less time for the actual data transmission, since data is transmitted after the end of the ATIM window, again degrading throughput at high loads. Large ATIM windows can also result in higher energy consumption since all nodes remain awake during the ATIM window. In particular, at a low load, large ATIM windows are unnecessary. Thus, a static ATIM window size cannot always perform well. This paper proposes a mechanism for choosing an optimal ATIM window size. Similar work has been done in [4], but in this paper, we propose a much simpler mechanism that achieves the same goal.

The rest of the paper is organized as follows. Section II reviews the related work and Section III presents our proposed approach. Section IV describes our simulation model and discusses the simulation results. Section V concludes the paper.

II. Related Work

Woesner *et al.*[11] presented simulation results for the power saving mechanisms of two wireless LAN standards, IEEE 802.11 and HIPERLAN. They showed the different sizes of beacon intervals and ATIM windows in IEEE 802.11 have significant impact on throughput and energy consumption. Tseng *et al.*[10] proposed some solutions for synchronizing beacon intervals when using DCF in multi-hop wireless networks.

Jung and Vaidya [4] proposed an energy efficient MAC protocol, where nodes increase and decrease their ATIM window sizes based on several rules. In this paper, we propose a similar approach, but with a much simpler mechanism that achieves the same goal. Another variation of IEEE 802.11 PSM is presented in [2].

SPAN [1] and GAF [12] are power saving techniques that require a group of nodes to stay awake all the time in order to forward traffic on behalf of dozing nodes.

In PAMAS [7], nodes use two separate channels – one for control packets and the other for data packets. Using the control channel, nodes determine when to power off and for how long. Similar to PAMAS, S-MAC [13] allows nodes to sleep during neighbors' transmissions; nodes go to sleep after hearing an RTS or a CTS destined for neighbors. Designed for wireless sensor networks, S-MAC reduces contention latency by *message passing*. Long messages are fragmented into many small fragments, which are transmitted in bursts. Only one RTS and one CTS are used to transmit many fragments. When nodes overhear an RTS or a CTS, they will go to sleep for the time that is needed to transmit all the fragments.

III. Proposed IPSM Scheme

We now present our proposed scheme, IPSM (Improved Power Saving Mechanism). IPSM has two major differences from PSM. First, in IPSM, a node can adjust its ATIM window size based on observed network conditions. Second, IPSM allows a node to enter the doze state in the middle of beacon interval if it has completed all the transmissions that are explicitly announced during the ATIM window. The latter is achieved by piggybacking the number of pending packets inside data packets.

In IPSM, a node uses only one ATIM frame to announce pending packets for the same destination during the same beacon interval. When the node transmits the actual data packets it includes the number of packets still pending for the destination. This information allows the destination node to determine when it has received all the packets pending at the source node at the time of data transmission. If the source node could not deliver all pending packets that were previously announced to the destination and the current beacon interval expires, both the source and destination stay up in the next beacon interval. During the next beacon interval the source transmits the remaining packets without additional ATIM frame. After that they may enter the doze state if they have no other announced pending packets to transmit or receive.

We now define few terms that are used in the rest of the paper.

- *ATIMmin:* The minimum ATIM window size that a node can have. *ATIMmin* is set to 2 ms in our simulations.
- *ATIMmax:* The maximum ATIM window size that a node can have. *ATIMmax* is set to 26 ms.
- *ATIMinc:* The amount of time that a node can increase for the ATIM window at a time. *ATIMinc* is set to 2 ms.
- *CIT (Channel Idle Time):* Current channel idle time measured at the end of the ATIM window. During the ATIM window, each node measures how long the channel was idle continuously. Whenever the channel is busy the *CIT* value is reset to zero. Therefore, if the *CIT* value is greater than zero at the end of the ATIM window, it implies that the channel is idle at that time. We assume that nodes sense the same channel conditions.
- *CITThreshold (Channel Idle Time Threshold):* At the end of the ATIM window if *CIT* is greater than *CITThreshold* the channel is idle long enough to assume that no node intends to transmit an ATIM frame. Thus, in this case, a node does not need to increase its ATIM window size. *CITThreshold* is set to 128 time slots (2.56 ms) in our simulations. The reason why *CITThreshold* is chosen to be 128 time slots is described later.

Fig. 2 illustrates the IPSM scheme. At the beginning of each beacon interval, nodes transmit beacons, as described in IEEE 802.11. After that, each node begins with ATIM window size equal to *ATIMmin*. When the ATIM window finishes a node compares *CIT* with *CITThreshold*. If *CIT* is less than or equal to *CITThreshold* the node increases its ATIM window by *ATIMinc* and restarts the ATIM window for *ATIMinc*. For

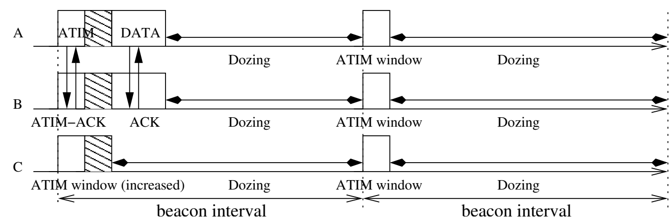


Fig. 2. IPSM: During the ATIM window node A and B transmit an ATIM and ATIM-ACK, respectively. At the end of the ATIM window, *CIT* is less than *CITThreshold*. Thus, nodes A, B, and C increase their ATIM window size by *ATIMinc*. When the increased ATIM window finishes nodes have *CIT* larger than *CITThreshold*, so A and B start their data transmission. Since C has no packet to send or receive it dozes after the ATIM window. Once completing all the data transmission A and B will also doze for the remaining beacon interval.

Algorithm 1 IPSM: ATIM Window Adjustment

```
ATIM window is set to ATIMmin;
finished := false;
repeat
  if ATIM window expired then
    Node checks its CIT;
    if CIT ≤ CITThreshold
    and ATIMsize < ATIMmax then
      Restart ATIM window for ATIMinc;
    else
      Start data transmission or enter the doze state;
      finished := true;
    end if
  end if
until finished is true;
```

instance, in Fig. 2, node A and B transmit an ATIM and ATIM-ACK, respectively, during the ATIM window. At the end of the ATIM window, A, B, and C have *CIT* less than *CITThreshold*. Thus, they increase their ATIM window size by *ATIMinc*. This process is repeated until *CIT* exceeds *CITThreshold*, or until the ATIM window size reaches *ATIMmax*. After the ATIM window finishes, nodes begin their data transmissions. When they complete their packet transmissions they enter the doze state. In our example, A and B start their data transmission after the increased ATIM window finishes. After completing data transmissions, they start dozing for the remaining beacon interval. Since C has no packet to send or receive, it dozes after the ATIM window. Pseudo code for adjusting the ATIM window is described in Algorithm. 1.

Note that there exists a finite delay associated with the doze-to-awake transition as well as a higher energy consumption. Therefore, in IPSM, a node will not enter the doze state after completing packet transmissions if the remaining duration in the current beacon interval is "too small" – specifically, in our simulations, a node will not enter the doze state if the remaining duration is less than 1600 μs – the delay for both doze-to-awake transition and awake-to-doze state transition are considered as 800 μs each [3], [6].

We now explain the reason why we set *CITThreshold* to 128 time slots. As mentioned earlier, each ATIM frame is transmitted using the CSMA/CA mechanism in IEEE 802.11. Specifically, a node wanting to transmit an ATIM frame picks a backoff interval in the range $[0, cw]$, where cw denotes the *contention window*. The initial cw value is called $cwmin$. The backoff interval is decremented by 1 after each "clock tick" if the channel is sensed as idle [9]. An ATIM frame is transmitted when the backoff interval reaches 0. If an ATIM-ACK is not received (possibly due to collisions) in response to the transmitted ATIM frame, the source node doubles the value of cw, selects a new backoff interval, and repeats the process to retransmit the ATIM frame. In IPSM, a node can retransmit an ATIM frame up to 3 times for the same destination. Therefore, when $cwmin$ is 31 the cw value can be doubled up to 128. This means that if the *CIT* value at the end of the ATIM window is larger than 128 time slots we can assume that no node has an ATIM frame to transmit. Therefore, nodes stop increasing their ATIM window sizes and start their data transmission when *CIT* is greater than *CITThreshold*.

IV. Performance Evaluation

We simulated the proposed IPSM scheme, the PSM scheme in IEEE 802.11, and IEEE 802.11 without any nodes in the power saving mode – we refer to the last one as WithOut Power Saving Mechanism (WOPSM). We use two metrics to evaluate the proposed scheme.

- *Aggregate throughput over all flows in the network*
- *Total data delivered per unit of energy consumption (Kbits delivered per joule)*: This metric measures the amount of data delivered per joule of energy. The total data delivered over all flows is divided by the total energy consumption over all nodes in the network. The total energy consumption is the sum of each node's energy consumption during the simulation time.

A. Simulation Model

We use ns-2 with the CMU wireless extensions [8] for our simulations. The duration of each simulation is 20 seconds in a wireless LAN. Each flow transmits CBR (Constant Bit Rate) traffic, and the rate of traffic is varied in different simulations. The channel bit rate is 2 Mbps and the packet size is fixed at 512 bytes.

For the energy model, we use 1.65 W, 1.4 W, 1.15 W, and 0.045 W to represent powers, consumed by the wireless network interface in the transmit, receive, and idle modes and the doze state, respectively. For a doze to awake transition time, we use 800 μs [3], [6], which is a much more conservative estimate than 250 μs in [5]. During this transition time, a node consumes twice the power of idle mode.

Each node starts with enough energy so that it will not run out of its energy during the simulations. All the simulation results are averages over 30 runs. We use 100 ms for each beacon interval [9]. For PSM simulation, we vary the ATIM window size from 2 ms to 40 ms. The number of nodes is chosen to be 10, 30 or 50. In all cases, half of the nodes transmit packets to the other half. Therefore, when the number of nodes is n there is a total of $\frac{n}{2}$ flows in the network.

With CBR traffic, we varied the total network load to observe the effect of network load on throughput and energy consumption. Simulated network loads are 10%, 20%, 40%, and 60%, measured as a fraction of the channel bit rate of 2 Mbps. For example, at network load of 10%, the total bit rate of all the traffic sources is $0.1 \times 2 = 0.2$ Mbps. Each traffic source has the same bit rate. Thus, with a total load of 10%, and with, say 10 traffic sources, each traffic source has a rate of 0.02 Mbps.

We also simulated each scheme with a dynamic network load using CBR. As Fig. 3 shows, the source nodes start with a network load of 40% and the network load changes from 40% to 10% at 5 seconds. The nodes then change from 10% to 20%, and from 20% to 30% at 5 second intervals.

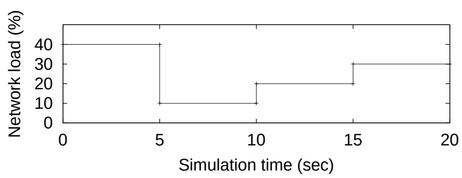


Fig. 3. In our dynamic network load scenario, the network load changes as the simulation time changes.

B. Simulation Results

1) Fixed Network Load: Fig. 4 shows the aggregate throughput (aggregated over all flows) with a fixed network load when using PSM, IPSM, and WOPSM schemes. In the figures, letter I on the horizontal axis corresponds to the proposed IPSM scheme, and W corresponds to the WOPSM scheme. The performance of the PSM scheme depends on the given ATIM window size. The figure plots the throughput of PSM as a function of the ATIM value, where the ATIM values are varied between 2 ms and 40 ms, as plotted on the horizontal axis. The results in Fig. 4 (a), (b) and (c) are for different numbers of nodes on the LAN.

As the graphs show, the ATIM window size impacts the throughput achieved by PSM quite significantly, especially with a large number of nodes in the network (Fig. 4 (c)). On the other hand, IPSM typically yields aggregate throughput comparable to WOPSM and PSM with the optimal ATIM window size.

Recall that in PSM (and IPSM), each node only transmits one ATIM frame for many pending packets for the same destination. Thus, with a small number of nodes in the network, the ATIM window size for PSM is less sensitive in Fig. 4 (a) as compared to that in Fig. 4 (c). For example, in Fig. 4 (a), a 4 ms ATIM window is enough to achieve the desirable throughput for a network load of 10%. However, with a network load of 40% in Fig. 4 (c), an ATIM window of about 15 ms gives the best throughput. As the number of nodes increases or the network load gets heavier, the ATIM window size becomes a significant factor for both throughput and the energy consumption in PSM. If the ATIM window is too small, there is not enough time to announce all the pending packets, resulting in throughput degradation. If the ATIM window is too large, there is less time for actual data transmission, resulting in throughput degradation as well. Aggregate throughput is also degraded in both PSM and IPSM with a network load of 60% in Fig. 4 (c). This is because the highly loaded network needs more time for data transmission, but both PSM and IPSM use the extra channel capacity for the ATIM window.

Fig. 5 shows the total data delivered per joule. IPSM performs better than PSM. Since with WOPSM all nodes are always awake, there is no energy savings from doze state. Although PSM allows a node to be in the power saving mode, the node cannot enter the doze state if it has at least one packet to transmit or receive. This is a disadvantage when the network load is low. In IPSM, a node can enter the doze state whenever it finishes the transmission. This allows node to be in the doze state longer than PSM, leading to reduced energy consumption as compared to PSM.

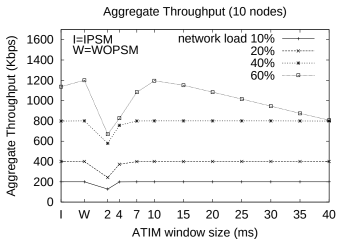
When the network load increases or the number of nodes increases energy saving from the power saving mode becomes smaller. For instance, the total data delivered per joule in the 30 and 50 node networks (Fig. 5 (b) and (c), respectively) are less than that in the 10 node network (Fig. 5 (a)). Note that the scales for the vertical axis in Fig. 5 (a), (b) and (c) are not identical. When the network load is high, there is less time for a node to be in the doze state due to data transmissions. Also, when the number of nodes in the network is large, there can be more collisions among the nodes. Therefore, a node will take more time to finish data transmission, hence, it is more likely to stay in the awake state. The shorter duration in the doze state will yield lesser energy conservation.

2) Dynamic network load: We now present results for the wireless LAN where the load is time-varying. The dynamic network load used in these simulations is shown in Fig. 3, as discussed previously. Fig. 6 plots the aggregate throughput for the IPSM scheme (label I on horizontal axis), the WOPSM scheme (label W on horizontal axis), and the PSM scheme (with different values of fixed ATIM window size for PSM). Fig. 7 shows the corresponding total data delivered per joule. The simulation results are similar to those for fixed network load.

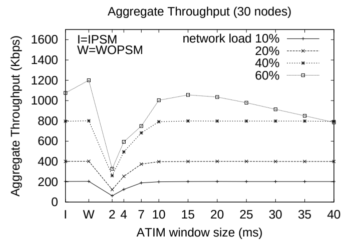
In Fig. 6 (a) and (b), ATIM windows of 5 ms and 10 ms in PSM gives comparable throughput to IPSM and WOPSM, respectively. However, the corresponding throughput per joule is lower than that of IPSM in Fig. 7 (a) and (b). As discussed earlier, since the ATIM window size affects both throughput and energy consumption, PSM does not perform well all the time. The ATIM window size should be dynamically adjusted according to the number of nodes contending in the network and the network load. The simulation using a dynamic network load shows that IPSM always performs well compared to PSM.

V. Conclusion

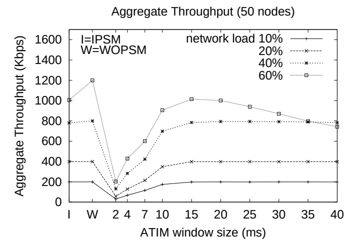
We have presented an energy efficient protocol that improves PSM (power saving mechanism) for DCF in IEEE 802.11. The ATIM window size is fixed in PSM, which is a significant factor for both throughput and the energy



(a) Aggregate throughput (10 flows)

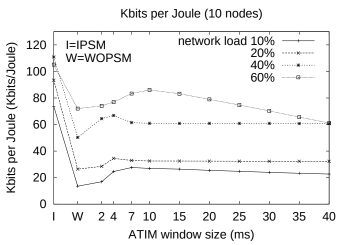


(b) Aggregate throughput (30 flows)

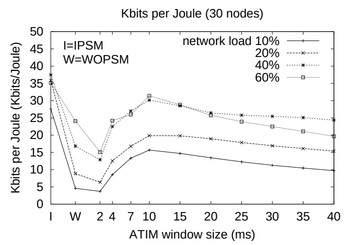


(c) Aggregate throughput (50 flows)

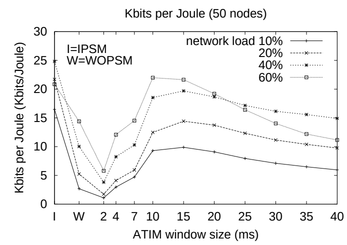
Fig. 4. Aggregate throughput: fixed network load.



(a) Total data delivered per joule (10 flows)

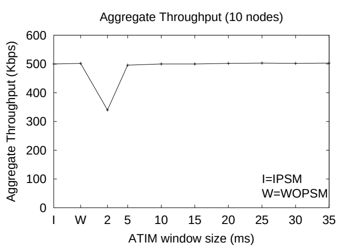


(b) Total data delivered per joule (30 flows)

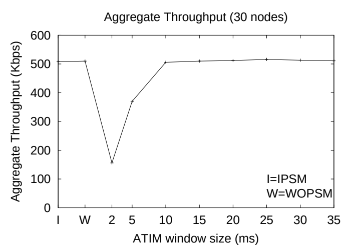


(c) Total data delivered per joule (50 flows)

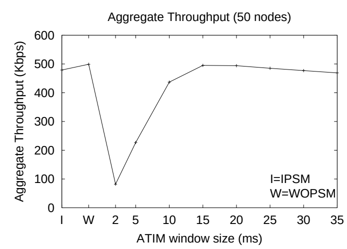
Fig. 5. Total data delivered per joule: fixed network load.



(a) Aggregate throughput (10 flows)

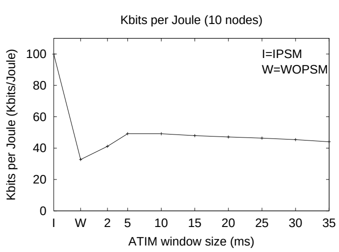


(b) Aggregate throughput (30 flows)

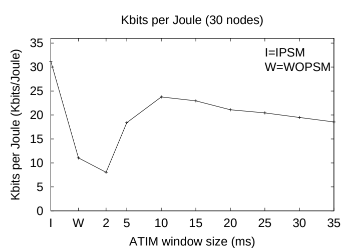


(c) Aggregate throughput (50 flows)

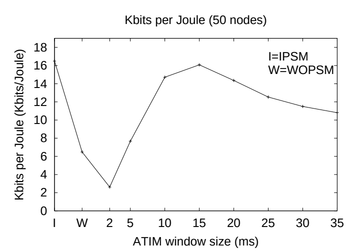
Fig. 6. Aggregate throughput: dynamic network load.



(a) Total data delivered per joule (10 flows)



(b) Total data delivered per joule (30 flows)



(c) Total data delivered per joule (50 flows)

Fig. 7. Total data delivered per joule: dynamic network load.

consumption. As the number of nodes increases or the network load gets heavier, the desirable ATIM window size becomes large. In IPSM, a node can adapt its ATIM window size based on observed network conditions. IPSM also allows node to power off its wireless network interface whenever it has finished transmitting packets. Simulation results show that IPSM outperforms PSM with respect to energy savings.

References

[1] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, “Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks,” in *MOBICOM 2001*, July 2001.

[2] J.-M. Choi, Y.-B. Ko, and J.-H. Kim, “Enhanced Power Saving Scheme for IEEE 802.11 DCF based Wireless Networks,” in *Personal Wireless Communications 2003*, September 2003.

[3] P. J. M. Havinga and G. J. M. Smit, “Energy-efficient TDMA Medium Access Control Protocol Scheduling,” in *Proc. Asian International Mobile Computing Conference (AMOC 2000)*, November 2000.

[4] E.-S. Jung and N. H. Vaidya, "An Energy Efficient MAC Protocol for Wireless LANs," in *INFOCOM 2002*, June 2002.
[5] A. Kamerman and L. Monteban, "WaveLAN-II: A High-Performance Wireless LAN for the Unilicensed Band," *Bell Labs Technical Journal*, vol. 2, summer 1997.
[6] T. Simunic, H. Vikalo, P. Glynn, and G. D. Micheli, "Energy Efficient Design of Portable Wireless Systems," in *International Symposium on Low Power Electronics and Design (ISLPED)*, July 2000.
[7] S. Singh, M. Woo, and C. S. Raghavendra, "Power-aware Routing in Mobile Ad Hoc Networks," in *MOBICOM '98*, October 1998.
[8] The CMU Monarch Project., "The CMU Monarch Project's Wireless and Mobility Extensions to NS."
[9] The editors of IEEE 802.11, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, 1997.
[10] Y.-C. Tseng, C.-S. Hsu, and T.-Y. Hsieh, "Power-Saving Protocols for IEEE 802.11-Based Multi-Hop Ad Hoc Networks," in *INFOCOM 2002*, June 2002.
[11] H. Woesner, J.-P. Ebert, M. Schlager, and A. Wolisz, "Power-Saving Mechanisms in Emerging Standards for Wireless LANs: The MAC Level Perspective," *IEEE Personal Communications*, June 1998.
[12] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed Energy Conservation for Ad Hoc Routing," in *MOBICOM 2001*, July 2001.
[13] W. Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," in *INFOCOM 2002*, June 2002.