A Power Efficient Scheduling Mechanism for an IEEE 802.11 Multiple Rate Ad Hoc Network

Weikuo Chu[1,2*] Yu-Chee Tseng[2]

[1]Department of Management Information
St. John's University
Tamsui, Taipei, Taiwan

wkchu@mis.sju.edu.tw

[2]Department of Computer Science
National Chiao-Tung University
Hsin-Chu, Taiwan

Received 19 February2008; Revised 6 June 2008; Accepted 10 June 2008

Abstract. Power management is one of the most important issues in mobile communications. Much research has been done in reducing wireless station's power consumptions. IEEE 802.11 addresses this issue by adopting a MAC layer active-doze Power Saving Mechanism. In an 802.11 ad hoc network, this Power Saving Mechanism works as follows. Any wireless station with data to send must first announce its traffic and then contends for the channel with other stations for data transmissions, all based on the DCF protocol. Stations not involved in any data transmissions can go to the doze mode to conserve energy. In this paper, we first show that this mechanism has the problem of power management inefficiency when used in a multiple rate ad hoc network. We then propose a novel scheduling mechanism, STFS, to reduce the total power consumptions of the wireless stations in the network. Simulation results show that the proposed scheduling mechanism does have better performance than that of 802.11 PSM.

Keywords: Ad Hoc Network, DCF, 802.11 PSM, Beacon Interval, and ATIM Window

1 Introduction

Wireless LAN or WLAN is the fastest growing field in mobile communications. By now, the majority of notebook computers and an increasing number of PDAs are equipped with wireless technology. Among the many wireless technologies, the family of IEEE 802.11 protocols is the most widely used access method for WLAN. In IEEE's proposed protocols for WLAN, there are two different ways to configure a network: ad hoc and infrastructure. In the ad hoc configuration, wireless stations (STAs) are brought together to form a network "on the fly". There is no structure to the network; there are no fixed points; and usually every STA is within the communication range of every other STA in the network. When configured in infrastructure mode, the WLAN consists of at least one access point (AP) connected to the wired network and a number of wireless STAs. The AP provides a local relay function for the network. All STAs in the network communicate with the AP and no longer communicate with each other directly.

In WLAN, battery power is an unavoidable issue that must be dealt with. In order to save power, 802.11 defines a MAC-layer Power Saving Mechanism (802.11 PSM) that allows a wireless STA to go from the active state to doze or power-saving state when the STA is not involved in any data transmissions [1]. In the infrastructure configuration of a WLAN, the AP will keep track of all STAs that are in power-saving state and buffer frames addressed to these STAs. These frames are kept until the STAs request them to be sent or discarded if they are not requested for a certain period of time. While in the case of ad hoc configuration, time is divided into Beacon Intervals and each Beacon Interval contains an ATIM (Ad Hoc Traffic Indication Message) Window followed by the Data Transmission Phase. The ATIM Window is used as the common awake period for all participating STAs to announce their traffic through ATIM frame transmissions. After the ATIM Window finishes, STAs that successfully send or receive ATIM frames must remain in the active state, and STAs can switch to power-saving state if they are not involved in any traffic announcements till the beginning of next ATIM Window. Actual data transfers occur in the Data Transmission Phase, and the normal DCF (Distributed Coordination Function) access procedure is used while sharing the transmission medium among the active STAs. Any STA that completes the ATIM frame transmission in the ATIM Window but fails to send data packet in the Data Transmission Phase will try to initiate another traffic announcement in the next ATIM Window. In addition to the 802.11 PSM, a number of power saving methods [2,3] covering all protocol layers from Physical to the Ap-

* Correspondence author

plication layer have also been proposed in the literature, and a system-level power-saving methodology for heterogeneous wireless networks is in [4].

Because of signal fading, interference, shadowing, and path loss, etc., wireless channels have time varying characteristics. As a result, different wireless STAs may perceive different channel qualities at the same time.

In order to obtain optimum throughput, STAs in the network need to use different transmission rates for different channel qualities [6]. But when 802.11 PSM is enabled in such a multiple rate ad hoc environment, we observe a problem of power management inefficiency which can be exemplified in Fig. 1. In this example, we assume there are 16 STAs in the network, 8 of which are transmitters (In this paper, a transmitter is an STA that only transmit, not receive data packets.), and 8 of which are receivers. Each transmitter has only one packet to send to its receiver and all data packets are equal in length. In those transmitters, 4 of them are fast STAs, and the other 4 are slow STAs. Since fast (slow) STAs will use less (more) time in sending packets, the packets transmitted by fast (slow) STAs are represented by narrow (wide) rectangles in Fig. 1. According to the operations of 802.11 PSM, these transmitters must first announce their traffic in the ATIM Window and then use DCF to contend for the channel in the Data Transmission Phase. In the worst case, it may happen that all slow transmitters win the channel contentions before any fast transmitter has a chance to send data packet. Therefore as shown in the upper half of Fig. 1, the numbers of STAs that must stay in the active/power-saving state in the first, second, and third Data Transmission Phases are 16/0, 12/4, and 8/8, respectively. That is, 4 of the 16 STAs must stay in the active state for 2 Beacon Intervals, and 8 STAs must remain active for all of the 3 Beacon Intervals. In order to save power, we will propose a scheduling mechanism called STFS (Shortest Time First Scheduling) in this paper so that the packets transmitted on the channel can be as shown in the lower half of Fig. 1. This scheduling mechanism has the characteristic that it will schedule all fast transmissions or transmissions using less time to proceed before any of the slow STAs is allowed to send packet in every Data Transmission Phase. By scheduling in this way, more STAs can complete their data transmissions earlier and then go to power-saving state to conserve energy. Now the numbers of active/power-saving STAs are only 16/0, 8/8, and 4/12 in Data Transmission Phases 1, 2, and 3, respectively, the total power consumptions of these STAs are thus minimized.

In the above example, we assume each transmitter only has a specified number of data packets to send, therefore after a transmitter completes all its data transmissions, it will go to the doze mode; that is, the number of active transmitters in each Beacon Interval may decrease over time. By scheduling fast transmissions to proceed first, STFS can make this decrease more significant, so more power can be saved.

The rest of the paper is organized as follows. Section 2 describes the operations of the proposed scheduling mechanism. The performance of the STFS is investigated in section 3 and conclusions are given in section 4.

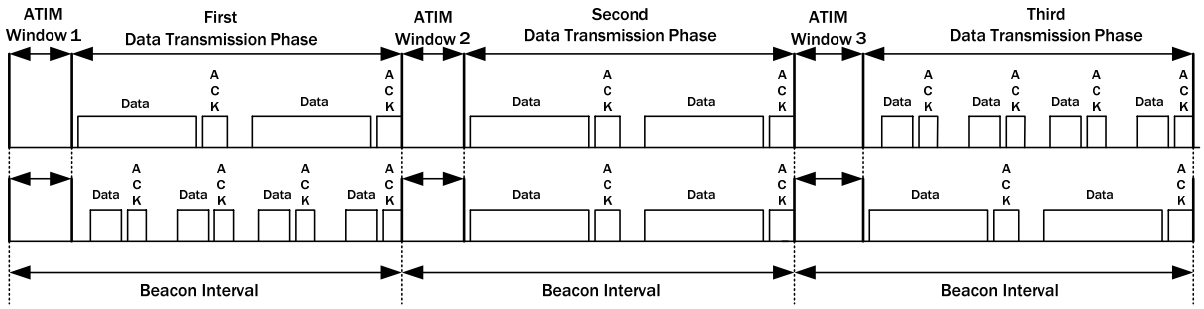


Fig. 1. The worst-case and best-case scenarios of power management in an 802.11 multiple rate ad hoc network..

2 The Shortest Time First Scheduling

In STFS, we assume: (1) The WLAN is configured in its ad hoc mode; (2) An ideal channel condition without packet losses is considered; (3) The Beacon Intervals begin and end approximately at the same time at all STAs, so the problem of time synchronization is not considered; (4) Each STA in the network can support k data rates, $r_1 > r_2 > \cdots > r_k$, and has implemented an automatic rate selection protocol such as the RBAR in [5] which enables a receiver to select the most appropriate rate for its sender to use in the Data Transmission Phase; (5) The data packets transmitted by all STAs are equal in length so the time required to transmit a packet is determined by its transmission rate; and (6) The promiscuous mode of the wireless interface is enabled so that the interface can intercept and read each network packet that arrives in its entirety.

As we mentioned earlier, STFS will schedule all fast transmissions before any of the slow transmissions in every Beacon Interval. A major problem with this scheduling mechanism is starvation; that is, some of the slow STAs may have no chances to send packets when Data Transmission Phase can not accommodate all active

transmissions. In order to achieve the goals of shortest time first and starvation prevention, we modify the packet formats of 2 control frames as follows: (1) The ATIM frame is extended with a 1-byte *aging* field; and (2) The ATIM-ACK is modified to include 2 additional 1-byte fields, *aging* and *rate*. The uses of these fields will be described in the following paragraph.

In addition to the above modifications, each STA in the network needs to maintain a local counter, *fc*. This counter has an initial value of 0. Whenever an STA has made a traffic announcement in an ATIM Window but fails to initiate transmission in the following Data Transmission Phase, *fc* is incremented by 1, otherwise *fc* is reset to 0. Before an ATIM frame is sent, the transmitter will copy the value of *fc* to the *aging* field of the frame. After an ATIM frame is received, the rate selected by the receiver is sent back to it's transmitter through the *rate* field of the ATIM-ACK. The contents of the field *aging* in ATIM-ACK are coming from the same field of the received ATIM frame.

For the purpose of deciding packet transmission order in every Data Transmission phase, a scheduling array of size q and a number of $2\cdot(k+1)$ indexes, s_0, e_0, s_1, e_1, ⋯, s_k, e_k, also need to be maintained by each STA in the network. The size of this array is such that it can accommodate at least k+1 non-overlapping queues, q_0, q_1, ⋯, and q_k; that is: $|q_0|+|q_1|+\cdots+|q_k|\le q$. The two ends, front and rear, of each q_i are pointed to by s_i and e_i, $0\le i\le k$, respectively. The configuration of these queues in the array is shown in Fig. 2. Whenever an STA receives an ATIM-ACK, the STA will use the *DA* (The Destination Address field, which now contains the address of the STA that transmitted the ATIM frame.), *rate*, and *aging* fields of the frame to update its scheduling array as follows: (1) If *aging* > 0, the contents of *DA* will be put into q_0; and (2) If *aging* = 0 and *rate* = r_i, the contents of *DA* will be put into q_i, $1\le i\le k$; that is, the addresses of all STAs with the local counter *fc*=0 and using the same data rate will be put into the same queue in the scheduling array. The order of the station addresses in queue q_i, $1\le i\le k$, is decided by the order of ATIM-ACK receptions, while the order in q_0 is determined as follows: The address in *DA* of ATIM-ACK$_1$ will have a smaller index value in q_0 than that in *DA* of ATIM-ACK$_2$ if (1) *aging* of ATIM-ACK$_1$ is larger than that of ATIM-ACK$_2$ or (2) *aging* of ATIM-ACK$_1$ is equal to that of ATIM-ACK$_2$ and *rate* of ATIM-ACK$_1$ is higher than that of ATIM-ACK$_2$ or (3) Both *aging* and *rate* of ATIM-ACK$_1$ are equal to those of ATIM-ACK$_2$ and ATIM-ACK$_1$ is received earlier than ATIM-ACK$_2$. For example, suppose an STA *X* receives 4 ATIM-ACKs with *DA*='*A*', *aging*=0, and *rate*=r_2 at time *t*, *DA*='*B*', *aging*=0, and *rate*=r_2 at time *t*+1, *DA*='*C*', *aging*=1, and *rate*=r_1 at time *t*+2, and *DA*='*D*', *aging*=2, and *rate*=r_2 at time *t*+3. Then, in the scheduling array of STA *X*, the address of STA *A* will have a smaller index value in q_2 than that of STA *B*, and the address of STA *D* will have a smaller index value in q_0 than that of STA *C*. When ATIM Window finishes, the array index values will be used by those STAs whose addresses are recorded in the scheduling array to setup the backoff counters to be used in data transmissions. Therefore all STAs whose addresses are in q_0 are permitted to send packets first, followed by the transmitters in q_1, and so on. Since the STAs whose addresses are in q_i will use a higher transmission rate than those whose addresses are in q_j, $1\le i<j\le k$, the goal of shortest time first is achieved. Any STAs that had completed traffic announcements but failed to transmit data in the previous Beacon Interval(s) are recorded in q_0, so the starvation problem mentioned above is also solved. After a transmitter completes its data transmission, it will reset its backoff counter value to e_k+1. This will give that transmitter chances to send multiple packets in the same Data Transmission Phase. After the current Beacon Interval terminates, the contents of the scheduling arrays maintained at all STAs are flushed to ensure the correct scheduling in the next Beacon Interval. A simple scheduling example of the STFS is shown in Fig. 3.

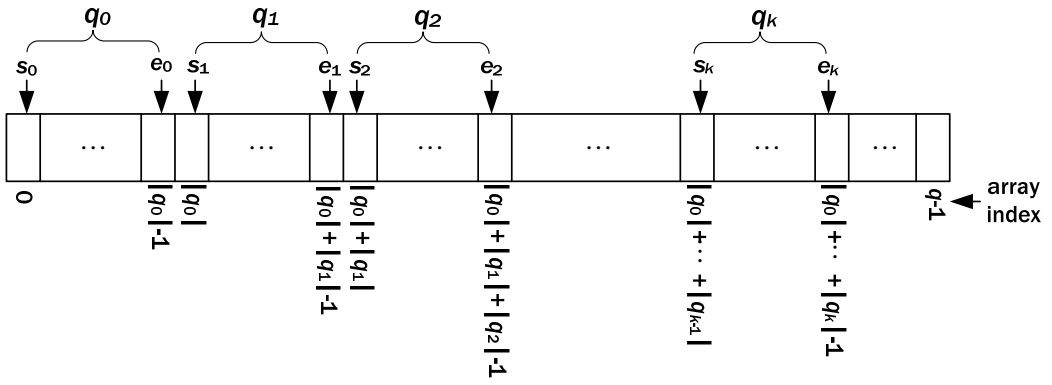
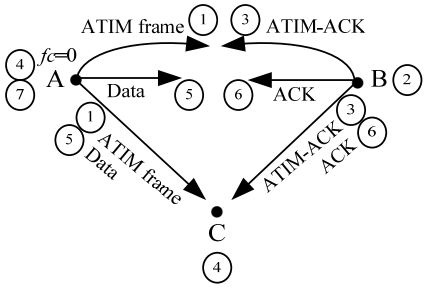


Fig. 2. The configuration of k+1 queues in the scheduling array.



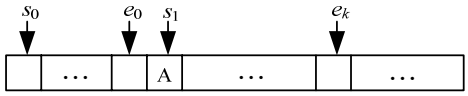
In ATIM Window:

Step1: A sends an ATIM frame with *aging* = 0 to B.
Since the promiscuous mode of the wireless interface has been enabled, this frame will also be received by C.

Step2: Based on the signal strength of the ATIM frame , B then selects the most appropriate rate, e.g. r_1, for A to use.

Step3: B sends the ATIM-ACK with *DA* = 'A', *rate* = r_1, and *aging* = 0 back to A. This ATIM-ACK will also be heard by C.

Step4: A and C extract the contents from *DA, aging,* and *rate* fields of the ATIM-ACK and use these information to update their respective scheduling arrays.
After the updates, the arrays at A and C may be like the following:



Station A will set its backoff counter to 6 in the following Data Transmission Phase.

In Data Transmission Phase:

Step5: After 6 idle time slots, A sends a data packet to B.

Step6: B responds with an ACK.

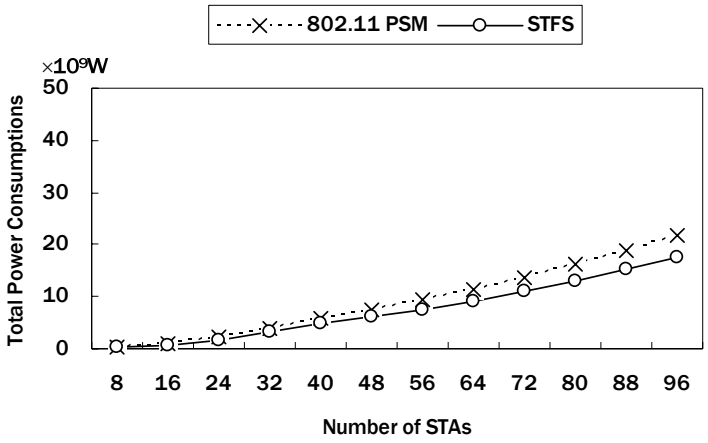
Step7: A resets its backoff counter to $e_k + 1 = 25 + 1 = 26$.

Fig. 3. A simple STFS scheduling example.

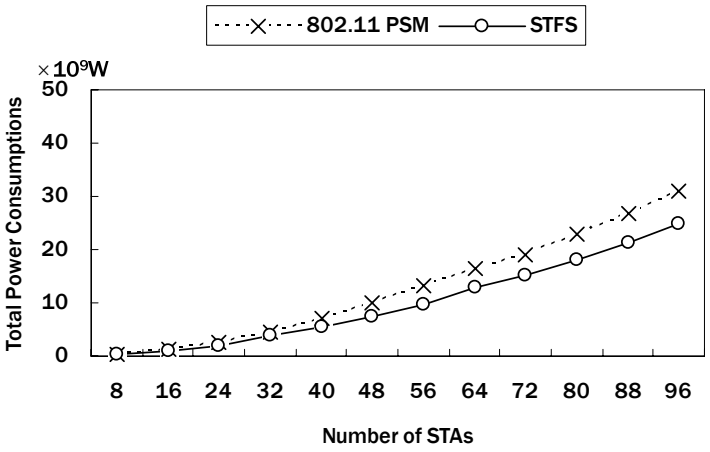
3 Performance of the STFS

We have developed a C++ based simulator to investigate the power consumption, channel usage, and throughput performance of the STFS and, for the purposes of comparison, the 802.11 PSM. Since the ATIM Window size will significantly affect the performance of 802.11 PSM [10,11], we will vary that size to be 30%, 40%, and 50% of the Beacon Interval in each set of the simulations to see its effect on the performance of STFS. In this paper, we assume an STA will never be both a transmitter and a receiver at the same time. An 802.11b-based ad hoc network is particularly considered in our simulations, so the STAs in the network can support k=4 different data rates, with r_1=11.0 Mbps, r_2=5.5 Mbps, r_3=2.0 Mbps, and r_4=1.0 Mbps. The rate used to send all control frames is 1 Mbps. In all simulations, we assume the numbers of transmitters that will use rate r_i, $1 \le i \le 4$, for data transmissions are equally distributed among all transmitters in the network. The size of the scheduling queue maintained at each STA is set to q=63. The packet size at the MAC layer is fixed at 1024 bytes, and the lengths of the Beacon, ATIM, and ATIM-ACK frames for 802.11 PSM are 50, 28, and 14 bytes, respectively. The Beacon Interval is set to be 100 ms. For the energy model, a wireless STA will consume 1.65 W, 1.4 W, 1.15 W, and 0.045 W in the transmit, receive, idle, and the power-saving states, respectively [7,8]. As in [9], the energy consumption for switching between awake and power-saving states is not considered in this paper. All simulation results are averages over 30 runs.

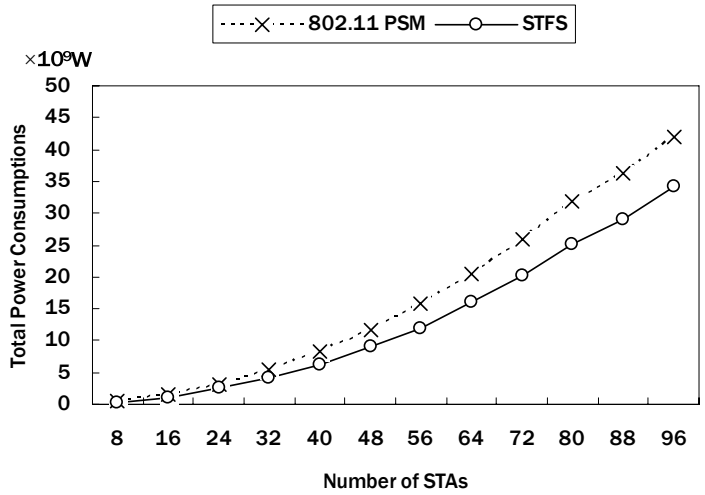
In our simulations, we measure the total power consumptions of all STAs for the case in which one half of the STAs are transmitters and each transmitter has 1000 data packets to send to its receiver. The results are shown in Fig. 4(a)~(c). As we can see from the results, the total power consumed by all STAs in the network is less in STFS than in 802.11 PSM for all situations. The percentage improvement on total power consumptions, defined as ($\text{TotalPowerConsumption}_{802.11PSM}$ - $\text{TotalPowerConsumption}_{STFS}$) / $\text{TotalPowerConsumption}_{802.11PSM}$, is shown in Fig. 5(d). We find a 20% to nearly 40% saving on energy is achieved by STFS. Finally, the results in Fig. 4(e) show that the savings on power consumption are more significant when the number of STAs in the network gets higher or the ATIM Window size becomes larger (When ATIM Window size gets larger, the Data Transmission Phase will become shorter for Beacon Intervals with fixed length.). When these situations occur, more STAs will remain active in the same Data Transmission Phase, so the less chance they all can complete data transmissions. By scheduling fast transmissions first, STFS can send more packets in every Data Transmission Phase, therefore more STAs can complete their transmissions earlier and then go to power saving mode to conserve energy.



(a) ATIM Window size = 0.3 * Beacon Interval

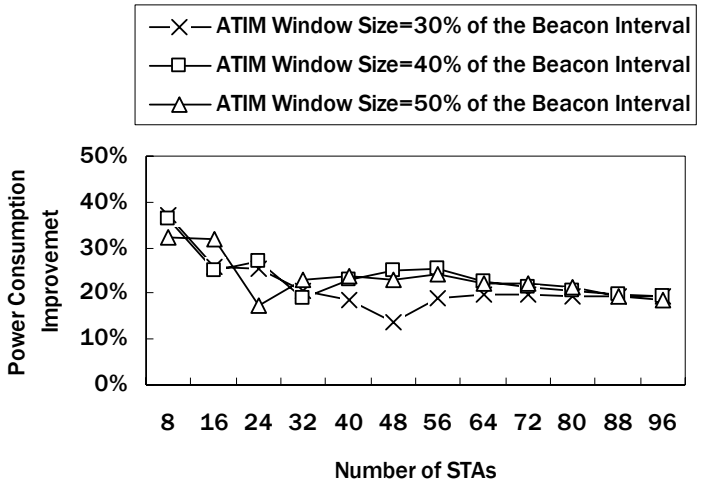


(b) ATIM Window size = 0.4 * Beacon Interval

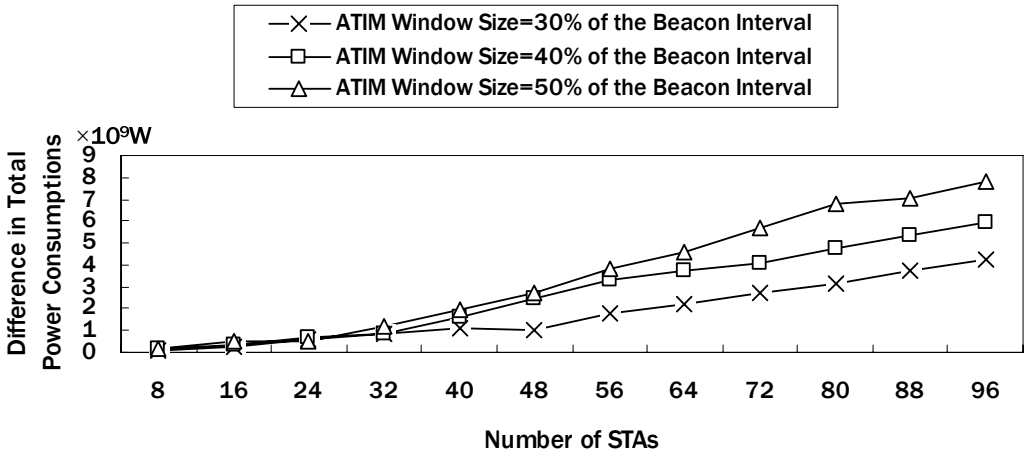


(c) ATIM Window size = 0.5 * Beacon Interval

Fig. 4. Power consumption performance of STFS and 802.11 PSM with each transmitter having 1000 data packets to send.



(d) Percentage Improvement on total power consumptions



(e) The difference in total power consumptions between STFS and 802.11PSM for ATIM Window of different sizes

Fig. 4. Power consumption performance of STFS and 802.11 PSM with each transmitter having 1000 data packets to send. (Cont.)

4 Conclusion

WLANs are usually designed for mobile applications. In mobile applications, battery power is one of the critical issues that must be dealt with. Due to limited battery power, various energy efficient protocols have been proposed to reduce wireless station's power consumptions in the literature. 802.11 addresses this power issue by allowing wireless stations to go into power-saving state at appropriate times to save power. However, this Power Saving Mechanism proposed by 802.11 has the problem of power management inefficiency when used in a multiple rate ad hoc network.

In this paper, a novel scheduling mechanism, STFS, is proposed to solve the above problem. The main idea of STFS is to schedule as many wireless stations to send packets as possible in every Beacon Interval so that they can complete their data transmissions earlier and then go to power-saving state to conserve energy. Simulation results show that the improvements made by STFS are significant and obvious in all situations.

References

[1] IEEE Std. 802.11-1999, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, 1999.

[2] H. Karl, *An Overview of Energy-Efficiency Techniques for Mobile Communication Systems*, Report of AG MOBICOM WG7, 2003.

[3] C. E. Jones, K. M. Sivalingam, P. Agrawal, and J. C. Chen, "A Survey of Energy Efficient Network Protocols for Wireless Networks," *Wireless Networks*, Vol. 7, No. 4, pp. 343-358, July 2001.

[4] T. Simunic, "Power Saving Techniques for Wireless LANs," *Proceedings of the Design, Automation, and Test in Europe Conference and Exhibition*, Vol. 3, pp. 96-97, 2005.

[5] G. Holland, N. Vaidya, and P. Bahl, "A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks," *Proceeding of ACM MOBICOM*, pp. 236-251, 2001.

[6] S. Ci, H. Sharif, "A Variable Data Rate Scheme to Enhance Throughput Performance of Wireless LANs," *Proceeding of IEEE CSNDSP*, 2002.

[7] Lucent, *IEEE802.11 WaveLAN PC Card - User's Guide*, page A-1.

[8] M. Stemm and R. H. Katz, "Measuring and Reducing Energy Consumption of Network Interfaces in Hand-Held Devices," *IEICE Transactions on Communications, special Issue on Mobile Computing*, Vol. E80-B, No. 8, pp.1125-1131, 1997.

[9] D. -Y. Kim and C.-H. Choi, "Adaptive Power Management for IEEE 802.11-Based Ad Hoc Networks," *Proceedings of the 5th World Wireless Congress*, 2004.

[10] 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*, Vol. 5, No. 3, pp. 40-48, 1998.

[11] E. -S. Jung and N. H. Vaidya, "An Energy Efficient MAC Protocol for Wireless LANs," *Proceedings of INFOCOM*, Vol. 3, pp. 1756- 1764, 2002.