Energy-Efficient Wireless Information Retrieval

Yu Jiao

Computational Sciences and Engineering Division

Oak Ridge National Laboratory

jiaoy@ornl.gov

Telephone: 1-865-574-0647

Fax: 1-865-576-0003

Ali R. Hurson

Department of Computer Science and Engineering

The Pennsylvania State University

hurson@cse.psu.edu

Telephone: 1-814-865-9505

Fax: 1-814-865-3176

Abstract

In this paper, we propose an adaptive application-driven power management (AADPM) protocol for wireless information retrieval applications within the IEEE 802.11b infrastructure WLAN environment. Our goal is to minimize energy consumption while achieving low round trip time delay. We discuss the protocol and evaluate its effectiveness using the network simulator NS2. We also draw horizontal comparisons among a variety of PM methods reported in the literature. Experimental results show that, compared to the power save mode supported by 802.11b, AADPM reduces the network interface card energy consumption by 52% while only introducing 3% RTT delay.

Keywords: IEEE 802.11b, power management, mobile computing, information retrieval

1. Introduction

The proliferation of Internet search engines and digital libraries has enabled effective data and knowledge exchange around the world, and they have become an indispensable part of our everyday life. The wide deployment of wireless networks and the more and more affordable portable devices give us the freedom to access the vast information reservoir from anywhere, at any time. However, battery-powered systems are often confronted with the problem of delivering high performance with a limited power supply. When using the wireless network, the network interface card (NIC) may consume up to 10% of the total energy of a high-end portable computer and nearly 50% of the overall energy of a low-end handheld devices [12]. Thus, NIC energy conservation is critical to prolong battery life.

In this paper, we use the IEEE 802.11 infrastructure wireless LAN as the target mobile environment. We propose an adaptive application-driven (AAD) power management (PM) protocol that has online learning capability. Our goal is to minimize the NIC energy consumption while achieving high throughput for information retrieval applications. We evaluate the effectiveness of the proposed scheme by drawing horizontal comparisons among a variety of PM strategies reported in the literature. We also investigate the impact of different parameters on the performance of AADPM.

The rest of the paper is organized as follows: section 2 presents the background and related work. Section 3 analyzes the NIC characteristics. Section 4 discusses the adaptive application-driven power management architecture and the protocol. Section 5 evaluates the proposed strategy and compares it with other PM methods. Finally, section 6 concludes our work.

2. Background and Related Work

The target network is the IEEE 802.11 infrastructure wireless LAN (as opposed to the IEEE 802.11 ad hoc network) [7], in which all stations communicate with an access point (AP). When the IEEE 802.11b power save mode (PSM) is enabled, the AP broadcasts a *Beacon* every *BeaconPeriod* (typically 100 ms). During the BeaconPeriod, the AP buffers all data destined for the stations. Each Beacon contains a traffic indication map (TIM) that shows whether a station has data buffered at the AP. IEEE 802.11b compliant wireless NICs support two power states: *awake* and *doze*. The NIC is fully powered while awake. In the doze state, the NIC is not able to transmit or receive and consumes very low power. When PSM is disabled, the NIC is in the continuous awake mode (CAM) and the AP forwards data packets destined for the mobile device without delay. With the PSM enabled, a mobile station can switch the NIC to doze state when it has no data to send or receive in order to conserve energy. The station wakes up the NIC periodically to listen to the Beacon. If the TIM indicates that there are data buffered at the AP, the mobile station must poll the AP and receive all outstanding data. Typically, a station listens to every Beacon, but it can also be configured to skip Beacons. Since its listening interval is fixed, this power management scheme is often referred to as PSM-static in the literature. Note that a mobile station cannot switch between CAM and PSM at will. It must notify the AP of the mode change intention. According to the 802.11b specification, a mode change can be carried out only after a successful frame exchange between the mobile station and the AP. A frame exchange typically includes a RTS (request-to-send) packet, a CTS (clear-to-send) packet, a data packet, and an ACK. Thus, it is not guaranteed that a mode change between CAM and PSM will always be successful, and the overhead introduced by such a transition depends on the network condition. It is reported that a mode transition may take as long as 600 ms [2].

Research has shown that, while PSM-static does quite well in saving energy in some cases, it is not flexible enough to balance the energy consumption and performance [2,11]. Many dynamic power management (DPM) strategies [2,4,6,11,12,15,16] have been developed to improve the performance of PSM-static. PM schemes are mainly proposed at three levels of a computer system: the hardware level (MAC or Link layer) [6,11], the operating system level [15], and the application level [2,4,12]. PM policies at each level often use one of the following techniques: fixed timeout, idle period prediction, or stochastic modeling.

An example of hardware-level timeout-based DPM is the Cisco Aironet 802.11a/b/g series of wireless LAN adapters. They provide a DPM option, which switches the NIC between the PSM and CAM depending on network traffic [6]. The DPM component switches the NIC to CAM when retrieving a large number of packets and switches it back to PSM after the retrieval. Researchers reported that the DPM switches the NIC to CAM when more than one packet is waiting at the AP, and it switches back to PSM after approximate 800 ms without receiving a packet [2]. We refer to this approach as the fixed-timeout DPM (FTDPM) in the rest of this paper.

Krashinsky and Balakrishnan proposed the Bounded-Slowdown (BSD) protocol [11], which operates completely at the link layer with no higher-layer knowledge. Under the assumption that NIC mode transitions between the CAM and PSM mode are always successful and introduce negligible overhead, the authors mathematically proved that the BSD protocol guarantees that the observed request round trip time (RTT) is less than ($1+p$) times longer than the RTT measured in the absence of PSM, where p is the maximum factor of RTT delay ($p > 0$). Experimental results show that in many cases the BSD protocol can significantly reduce the energy consumption of web page retrieval while guaranteeing the performance bound.

Stochastic models are often used to obtain optimal power management algorithms. Simunic et al. proposed a system-level time-indexed semi-Markov decision process (TISMDP) model and showed a factor of three in power savings with a low performance penalty for WLAN Telnet applications [15]. Stochastic model-based DPM strategies share two common drawbacks: First, the optimal solution can only be achieved if the real application follows the distribution assumed by the stochastic model; second, the computational overhead is often large. For example, the TISMDP algorithm has a time complexity of $O(N)$, where N is the maximum time index. Each of the $O(N)$ states requires evaluations of one double and two single integrals.

Compared with low-level DPM strategies, application-driven DPM approaches have the advantage of knowing the application's intention of network usage. This knowledge enables more aggressive power saving methods without sacrificing the performance. However, unlike low-level DPM, there is no one-size-fits-all solution to application level DPM problems because applications may exhibit drastically different network traffic patterns. Many application-level DPMs have been developed to embrace the energy/performance tradeoff.

Self-tuning power management (STPM) [2] is a system-level strategy that utilizes application-level knowledge. It applies a concept similar to but more sophisticated than the hardware level DPM of the Cisco Aironet LAN adapter. If an application's delay tolerance is less than the maximum latency of PSM, or the number or size of the upcoming transfer(s) is large, STPM will put the NIC in CAM. When no transfers are in progress, no application has specified a tolerance of delay less than the PSM's maximum latency, and the estimated idle period is long, STPM will switch the NIC back to PSM. The estimations of upcoming transfers and idle periods are based on the history of a group of requests that are closely related in time (150 ms). Results show that STPM can reduce the energy usage of a distributed file system by 21% compared to PSM, while reducing the RTT delay by 80%. However, STPM is not suitable for request/response applications that involve long user-think time (e.g. web browsing) or applications that have long server response times (e.g. distributed information retrieval), because they will not

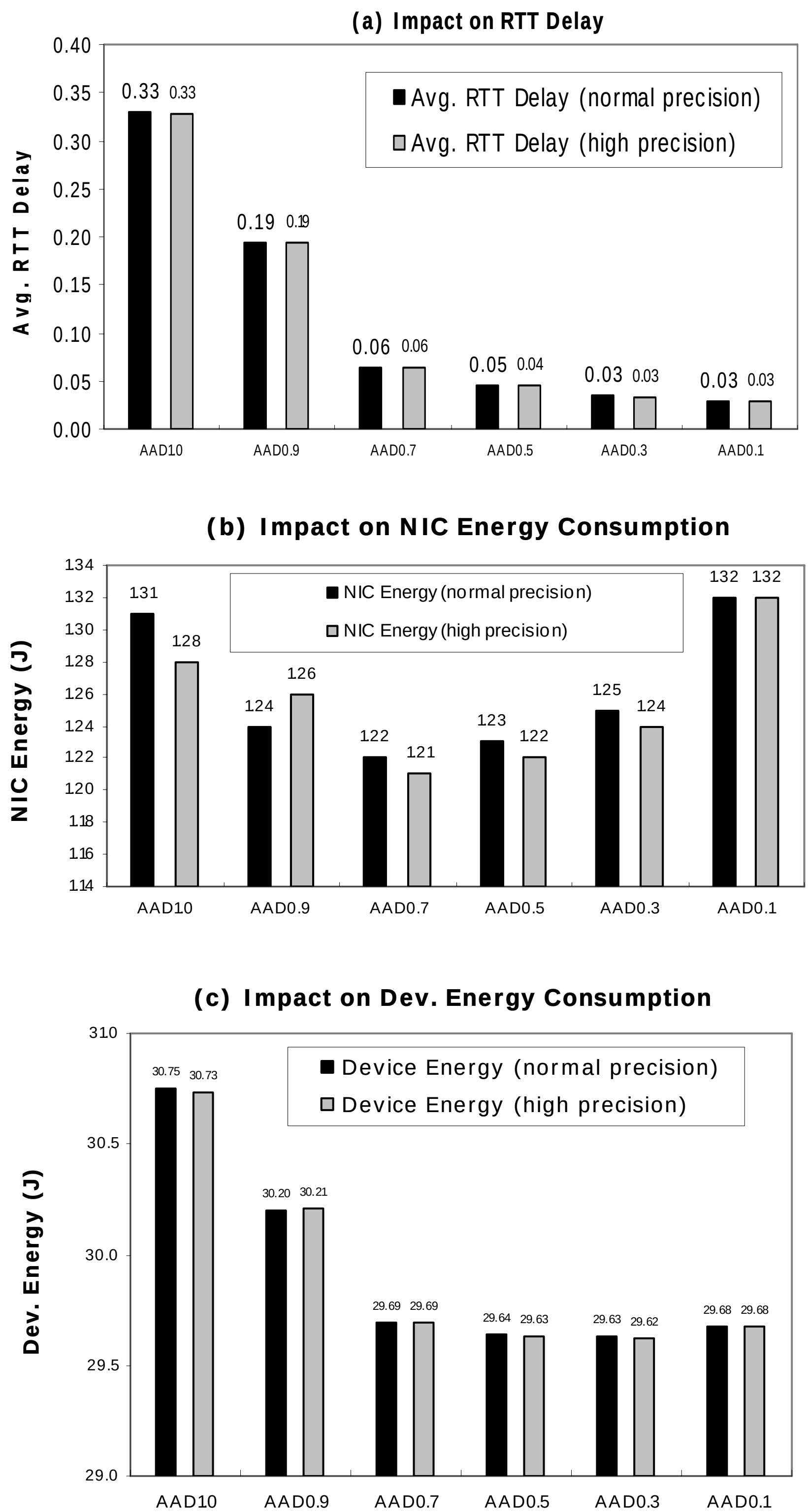


Figure 14. Impact of the precision of the histograms

Figure 14 shows the impact of the precision of the histograms on the RTT delay and the average NIC and device energy consumption per request. We can see that the performance gain from increasing the histogram size by a factor of 4 is minimal, if at all, with all ep_ratio settings. Thus, we conclude that the performance of AADPM cannot be improved significantly by increasing the histogram size.

5.4. Impact of AADPM History Window Size

In order to study the impact of the history window size (ω) on the AADPM performance, we vary ω from 10 to 100 while keeping all other parameters the same as those of subsection 5.2. Results indicate that AAD1.0 and AAD0.9 behave similarly, while AAD0.1, AAD0.3, AAD0.5, and AAD0.7 show strong resemblance. Thus, we only plot AAD0.9 and AAD0.3 in Figure 15.

As one can see from the above figures that there is no linear relationship between the history window size and the performance metrics – increased window size does not always result in the improvement of performance. The impact of ω also depends on the ep_ratio. When ω is increased from 10 to 20 (Figure 15a), the RTT delay of AAD0.3 is drastically reduced (87%), however, the delay of AAD0.9 is increased 5%. As we continue to raise the value of ω from 20 to 100, the RTT delay of AAD0.3 remains about the same, while AAD0.9 shows a gradual small increase in its RTT delay.

The variations of ω caused small fluctuations (< 0.5 J) in the average NIC (Figure 15b) as well as the overall device (Figure 15c) energy consumption per request for both AAD0.3 and AAD0.9.

We conclude that increasing the history window size alone cannot guarantee a better PM performance. Large history window size allows the PM component to observe longer history, while small window size reflects the situation changes more spontaneously. The value of ω should be chosen according to the nature of the application. Past traces can be used as training data to determine a suitable history window size for a particular application.

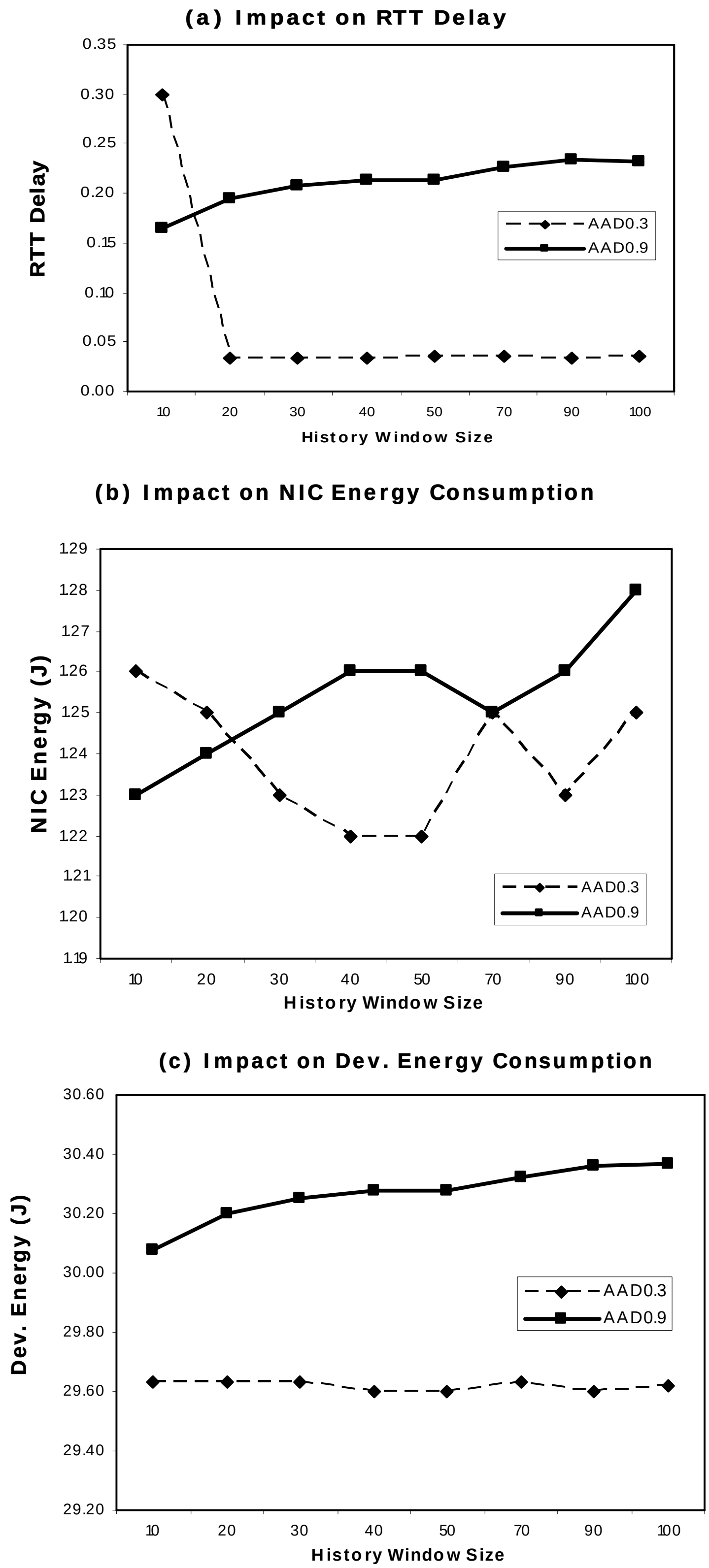
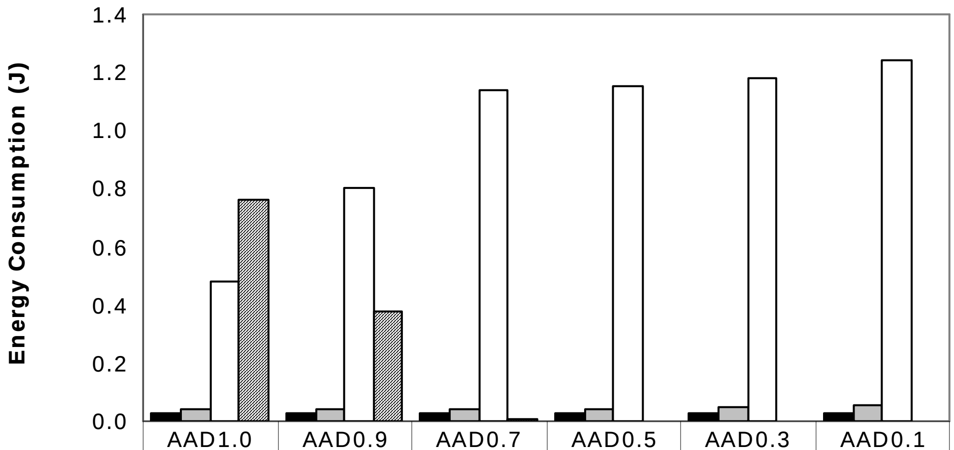


Figure 15. Impact of history window size

5.5. Impact of AADPM Predictive Wakeup

AADPM will suspend the NIC if it predicts that the upcoming idle period is longer than 18.97 s and by default, the NIC will remain suspended till the arrival of the next request. This means that once the NIC is suspended, it will be forced to wakeup when a request arrives and the request will encounter a delay of 600 ms. Figure 16 presents a decomposed view of the NIC energy consumption of AADPM with different ep_ratios.



	AAD1.0	AAD0.9	AAD0.7	AAD0.5	AAD0.3	AAD0.1
Transmit	0.03	0.03	0.03	0.03	0.03	0.03
Receive	0.04	0.04	0.04	0.04	0.05	0.05
Idle	0.48	0.80	1.14	1.16	1.18	1.24
Forced Wakeup	0.76	0.38	0.01	0.00	0.00	0.00

Figure 16. AADPM average NIC energy consumption per request

The forced wakeup energy in Figure 16 shows the average energy that the NIC consumed during the transition from being suspended to transmit. Results indicate that only when the ep_ratios are 1.0 and 0.9, are the suspended state of the NIC significantly utilized. This extra energy consumption also explains the noticeably longer RTT delays of AAD1.0 and AAD0.9 in Figure 9.

We can reduce both the RTT delay and the forced wakeup energy by predictive wakeup – during the user-think time, a suspended NIC will be reactivated (placed in CAM), once the predicted idle time has passed. We call the AADPM strategy with predictive wakeup AADPM+. Figure 17 compares the AAD1.0 with AAD1.0+ and AAD0.9 with AAD0.9+.

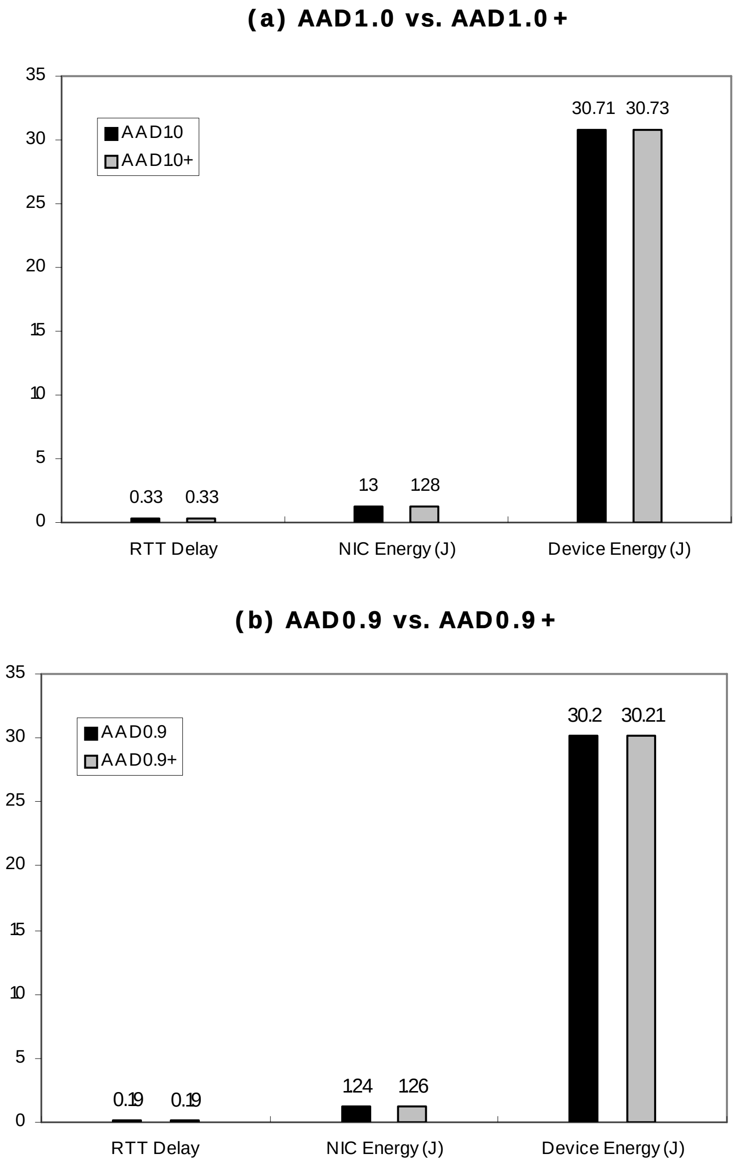
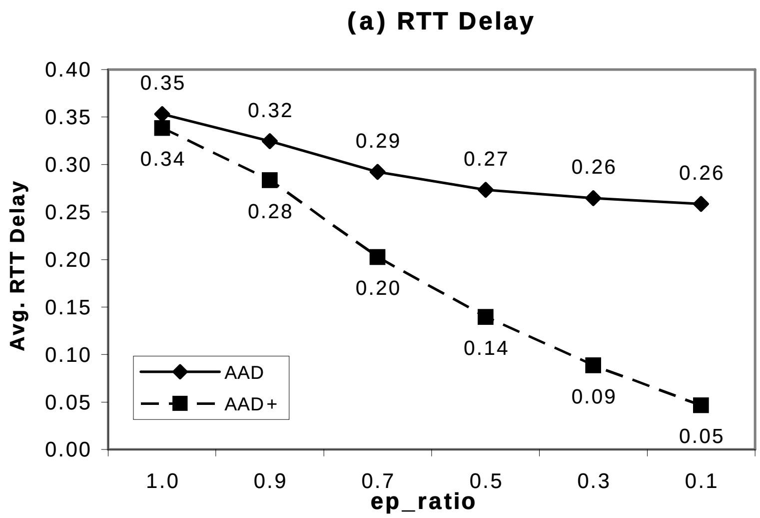
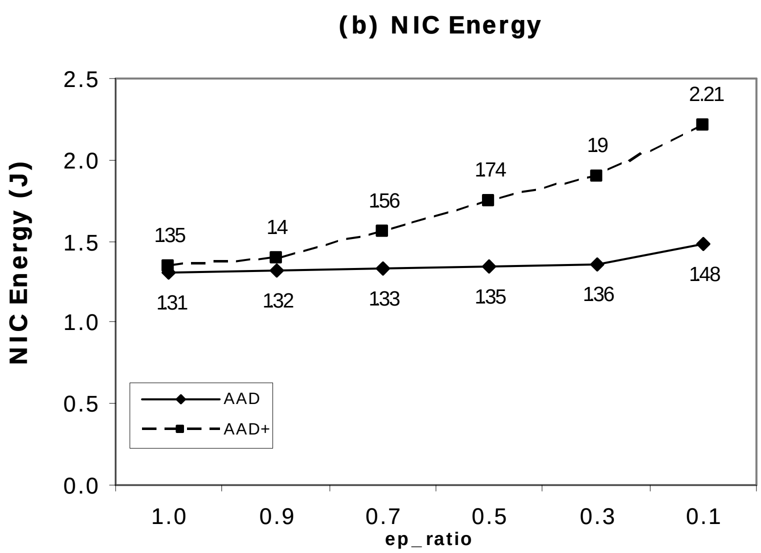


Figure 17. Impact of predictive wakeup

To our surprise, the predictive wakeup did not significantly improve the average RTT delay and NIC energy consumption, and it slightly worsens the over all device energy consumption. A close examination of the execution traces reveals that the reason is the user-think time distribution. Since we used a Weibull distribution with a location parameter of 15 s, only about 10% of the user-think times generated by the distribution are longer than 18.97 s. When the ep_ratios are 1.0 and 0.9, the AADPM component makes liberal predictions of the idle period that are often longer than the real values. Thus, the predictive wakeup scheme does not have significant impact on the performance metrics. As the location parameter shifts to a higher value, however, the predictive wakeup method demonstrates improved performance, as shown below. We increased the location parameter from 15 s to 20 s and repeated the above experiments. Figure 18 plots the results.





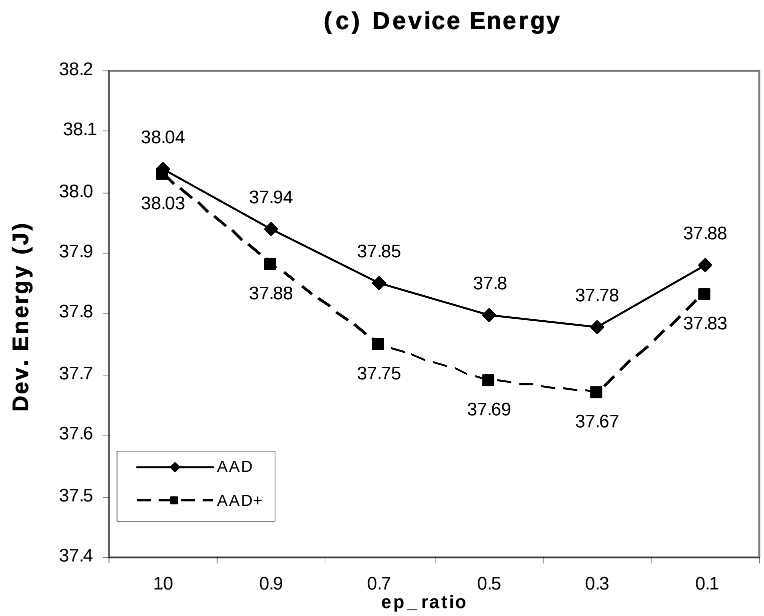


Figure 18. Impact of predictive wakeup with long user-think time

When the user-think time increases, the predictive wakeup method reduces the average RTT delay at the price of a higher NIC energy consumption. However, the average overall device energy consumption per request decreases under all ep_ratio configurations.

Each individual user may exhibit a different think time pattern, which can be used to determine whether or not to apply predictive wakeup. If the user profile indicates short think time (much shorter than the break-even time of the suspended state of the NIC), the AADPM predictive wakeup function should not be used. Otherwise, if a user tends to have long think times, predictive wakeup should be used to shorten RTT delays as well as to reduce the overall device energy consumption.

6. Conclusion

This paper analyzed the characteristics of distributed information retrieval applications and discussed the energy-efficiency issues of such applications in an IEEE 802.11b wireless LAN environment. We designed and evaluated an adaptive application-driven power management (AADPM) strategy, which can achieve high throughput while minimizing the NIC energy consumption.

We extended the network simulator NS2 to support the IEEE 802.11b power save mode and implemented a library of PM strategies that have been reported in the literature. We evaluated the proposed scheme through extensive simulation. We have also drawn horizontal comparisons among different PM methods in terms of the RTT delay, NIC energy consumption, overall mobile device energy consumption, and AP buffer size.

Our experimental results have shown that by knowing the application's intentions, AADPM achieved the best performance/ energy balance. Compared to NO_PM, AADPM (with ep_ratio 0.5) reduced the NIC energy consumption by 92% at the cost of a 5% RTT delay increase. When compared with PSM, AAD0.5 improved the RTT delay and energy saving by 87% and 52%, respectively. Results also indicate that low-level DPM strategies such as FTDPM and BSD can achieve good throughput by only conservatively putting the NIC to doze. However, the NIC energy consumption of these two dynamic methods is higher than the static PSM approach.

We also observed that if a PM strategy cannot balance the energy saving of the NIC and the RTT delay that it introduces, it may decrease the NIC energy consumption, but increase the overall mobile device energy consumption. We formally analyzed this problem in section 5.2 and an equation is given to determine whether or not a PM strategy is beneficial.

Histogram precisions, history window size, and the predictive wakeup technique have direct impact on the performance of the AADPM. Experiments show that simply increase the precision and the size of the history window size cannot guarantee shorter RTT delays and less energy consumption. Meanwhile, they increase the memory requirement. The predictive wakeup strategy is effective when the user-think time is often longer than the break-even time of the NIC's suspended state. It is recommended to take advantage of user profiling and use the past traces as the training data set in order to find out the optimal setting for these three parameters.

ADDPM is a powerful and easy-to-implement scheme. As part of our future research, we are working on implementing ADDPM on a personal digital assistant.

7. References

[1] Agere Systems Inc., *Wireless LAN PC Card (Extended)*, available from http://www.agere.com/client/docs/DS02115-1.pdf.

[2] M. Anand, E.B. Nightingale, and J. Flinn, "Self-Tuning Wireless Network Power Management", *Proc. of ACM MOBICOM*, San Diego, CA, September 2003, pp. 176-189.

[3] L. Benini, A. Bogliolo, and G.D. Micheli, "A Survey of Design Techniques for System-Level Dynamic Power Management", *IEEE Trans. on VLSI Systems*, 8(3), 2000, pp. 299-316.

[4] S. Chandra, "Wireless Network Interface Energy Consumption Implications of Popular Streaming Formats", *Proc. of Multimedia Computing and Networking (MMCN)*, San Jose, CA, January 2002, pp. 85-99.

[5] J.-C. Chen, K.M. Sivalingam, and P. Agrawal, "Performance Comparison of Battery Power Consumption in Wireless Multiple Access Protocols", *ACM Wireless Networks*, 5(6), 1999, pp.445-460.

[6] Cisco Systems, Inc., *Cisco Aironet 802.11a/b/g Wireless LAN Client Adapters Installation and Configuration Guide*, 2001.

[7] IEEE Computer Society LAN MAN Standards Committee, *IEEE Std 802.11: Wireless LAN Medium Access Control and Physical Layer Specifications*, August 1999.

[8] S. Irani, S. Shukla, and R. Gupta, "Online Strategies for Dynamic Power Management in Systems with Multiple Power-Saving States", *ACM Trans. on Embedded Computing Systems*, 2(3), August 2003, pp. 325-346.

[9] Y. Jiao and A.R. Hurson, "Mobile Agents in Mobile Data access Systems", *Proc. of the 10th Intl. Conf. on Cooperative Information Systems (COOPIS)*, 2002.

[10] C.E. Jones, K.M. Sivalingam, P. Agrawal, and J.-C. Chen, "A Survey of Energy Efficient Network Protocols for Wireless Networks", *ACM Wireless Networks*, 7(4), July 2001, pp. 343-358.

[11] R. Krashinsky and H. Balakrishnan, "Minimizing Energy for Wireless Web Access with Bounded Slowdown", *Proc. of ACM MOBICOM*, Atlanta, GA, July 2002, pp. 119-130.

[12] R. Kravets and P. Krishnan, "Application-Driven Power Management for Mobile Communication", *ACM Wireless Networks*, 6(4), 2000, pp.263-277.

[13] B.A. Mah, "An Empirical Model of http Network Traffic", *Proc. of IEEE INFOCOM*, Kobe, Japan, April 1997.

[14] X.Q. Meng, S. H. Y. Wong, Y. Yuan, and S.W. Lu, "Characterizing flows in large wireless data networks", *Porc. Of ACM MOBICOM*, Philadelphia, PA, September 2004, pp. 174-186.

[15] T. Simunic, L. Benini, P. Glynn, and G. De Micheli, "Event-Driven Power Management", *IEEE Trans. On Computer-Aided Design of Integrated Circuits and Systems*, *20*(7), pp. 840-857.

[16] E. Shih, P. Bahl, and H. Balakrishnan, "Wake on Wireless: An Event –Driven Energy Saving Strategy for Battery Operated Devices", *Proc. of ACM MOBICOM*, Atlanta, GA, July 2002.

[17] The VINT Project, *The NS Manual*, November 2001.

[18] UC Berkeley Home IP Web Traces, available from http://ita.ee.lbl.gov/html/contrib/UCB.home-IP-HTTP.html.