Power Management Strategies for Mobile Devices in Wireless Networks

Kyungseo Park
Department of Computer Science & Engineering
University of Texas at Arlington
kpark@cse.uta.edu

Abstract

As wireless networks become an integral component of the modern communication infrastructure, energy efficiency will be an important design consideration. Since batteries provide limited power, a general constraint of wireless communication is the short continuous operation time of mobile terminals. Therefore power management is one of the most challenging problems in wireless communication, and recent research has addressed this topic. Basically, IEEE 802.11 introduces a basic idea of power saving mode (PSM). In this paper, a couple of ideas are introduced to save energy in the infrastructure and ad hoc architecture of wireless communications. The first one is the modification of using two different power levels for RTS-CTS and DATA-ACK based on ad hoc network. This modification does not degrade throughput and yields energy saving. The second one is the Bounded Slowdown (BSD) protocol, which is the modification of PSM and based on infrastructure.

1. Introduction

The capabilities of mobile computing devices are often limited by the size and lifetime of the batteries that power them. As a result, minimizing the energy usage of every component in a mobile system is an important design goal. Wireless network access is a fundamental enabling feature for many portable computers, but if not optimized for power consumption, the wireless network interface can quickly drain a device's batteries.

Studies show that the significant consumers of power in a typical laptop are the microprocessor (CPU), liquid crystal display (LCD), hard disk, system memory (DRAM), keyboard/mouse, CDROM drive, floppy drive, I/O subsystem, and the wireless network interface card [5,6]. A typical example from a Toshiba 410 CDT mobile computer demonstrates that nearly 36% of power consumed is by the display, 21% by the CPU/memory, 18% by the wireless interface, and 18% by the hard drive. Consequently, energy conservation has been largely considered in the hardware design of the mobile terminal [7] and in components such as CPU, disks, displays, etc. Significant additional power savings may result by incorporating low-power strategies into the design of network protocols used for data communication. Many wireless network interfaces, especially wireless LAN cards, consume a significant amount of energy not only while sending and receiving data, but also when they are idle with their radios powered up and able to communicate [8].

In the so-called infrastructure mode , a mobile device communicates with a wired access point (AP). When 802.11 PSM is enabled, the AP buffers data destined for the device [9]. Once every *BeaconPeriod*, typically 100 ms, the AP sends a beacon containing a traffic indication map (TIM) that indicates whether or not the mobile device has any data waiting for it. The mobile device wakes up to listen to beacons at a fixed frequency and polls the AP to receive any buffered data. Typically, it listens to every beacon, but the mobile device can also be configured to skip *ListenInterval* beacons between listen times. Whenever the AP sends data to the mobile device, it indicates whether or not there is more data outstanding, and the mobile device goes to sleep only when it has retrieved all pending data from the AP. When the mobile device itself has data to send, it can wake up to send the data without waiting for a beacon. The 802.11 PSM is an example of a static power-saving algorithm, since it does not adapt the sleep and awake durations to the degree of network activity; it will be referred as PSM-static in this paper. It is found that while PSM-static does quite well in saving energy, it does so at significant performance cost. PSM-static can have an especially adverse impact on short TCP connections, whose performance is dominated by the connection round trip time(RTT). And an interesting inversion effect can occur, where under some conditions, the time to transfer a file over a wireless network running PSM-static increases when the bandwidth of the wireless link increases. Furthermore, with PSM-static, the power consumed while sleeping and listening for beacons dominates the total energy consumption if the network is accessed only sporadically.

Meanwhile, a simple power control protocol has been proposed based on an RTS-CTS handshake in the context of IEEE 802.11. Different power levels among different nodes introduce asymmetric links. Therefore, RTS and CTS are transmitted using the highest power level and DATA and ACK are transmitted using the minimum power level necessary for the nodes to communicate. This scheme has a shortcoming, which increases collisions and degrades network throughput. A new power control MAC protocol(PCM) is presented, which does not degrade throughput.

2. Background

2.1 Sources of Power Consumption

The sources of power consumption, with regard to network operations, can be classified into two types: communication related and computation related. Communication involves usage of the transceiver at the source, intermediate (in the case of ad hoc networks), and destination nodes. A typical mobile radio may exist in three modes: transmit, receive, and standby. Maximum power is consumed in the transmit mode, and the least in the standby mode. For example, the Proxim RangeLAN2 2.4 GHz 1.6 Mbps PCMCIA card requires 1.5 W in transmit, 0.75 W in receive, and 0.01 W in standby mode. In addition, turnaround between transmit and receive modes (and vice-versa) typically takes between 6 and 30 microseconds. Power consumption for Lucent's 15 dBm 2.4 GHz 2 Mbps Wavelan PCMCIA card is 1.82 W in transmit mode, 1.80 W in receive mode, and 0.18 W in standby mode. Thus, the goal of protocol development for environments with limited power resources is to optimize the transceiver usage for a given communication task. The computation considered is chiefly concerned with protocol processing aspects. It mainly involves usage of the CPU and main memory and, to a very small extent, the disk or other components. Also, data compression techniques, which reduce packet length (and hence energy usage), may result in increased power consumption due to increased computation. There exists a potential tradeoff between computation and communication costs. Techniques that strive to achieve lower communication costs may result in higher computation needs, and vice-versa. Hence, protocols that are developed with energy efficiency goals should attempt to strike a balance between the two costs [3].

2.2 Wireless network Architecture

• Infrastructure

Wireless networks often extend, rather than replace, wired networks, and are referred to as infrastructure networks. A hierarchy of wide area and local area wired networks is used as the backbone network. The wired backbone connects to special switching nodes called base stations. Base stations are often conventional PCs and workstations equipped with custom wireless adapter cards. They are responsible for coordinating access to one or more transmission channel(s) for mobiles located within the coverage cell. Transmission channels may be individual frequencies in FDMA (Frequency Division Multiple Access), time slots in TDMA (Time Division Multiple Access), or orthogonal codes or hopping patterns in the case of CDMA (Code Division Multiple Access). Therefore, within infrastructure networks, wireless access to and from the wired host occurs in the last hop between base stations and mobile hosts that share the bandwidth of the wireless channel.

•Ad hoc

Ad hoc networks, on the other hand, are multihop wireless networks in which a set of mobiles cooperatively maintain network connectivity [4]. This on-demand network architecture is completely un-tethered from physical wires. Ad hoc networks are characterized by dynamic, unpredictable, random, multi-hop topologies with typically no infrastructure support. The mobiles must periodically exchange topology information which is used for routing updates. Ad hoc networks are helpful in situations in which temporary network connectivity is needed, and are often used for military environments, disaster relief, and so on.

2.3 802.11 MAC Power Saving Background

The IEEE 802.11 standard recommends the following technique for power conservation. A mobile that wishes to conserve power may switch to sleep mode and inform the base station of this decision. The base station buffers packets received from the network that are destined for the sleeping mobile. The base station periodically transmits a beacon that contains information about such buffered packets. When the mobile wakes up, it listens for this beacon, and responds to the base station which then forwards the packets. This approach conserves power but results in additional delays at the mobile that may affect the quality of service (QoS) [3].

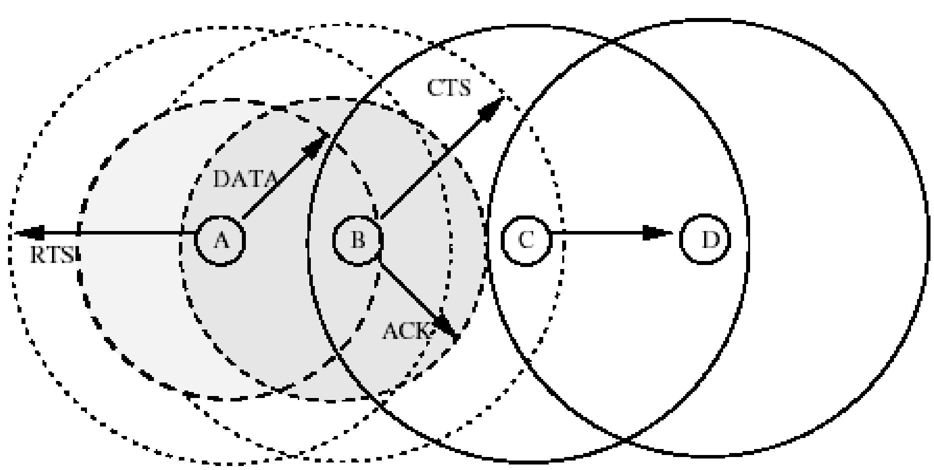


Figure 1. BASIC schemes

3. A Power Control MAC Protocol [2]

3.1 Basic Protocol Description

Power control can reduce energy consumption. However, power control may introduce different transmit power levels at different hosts, creating an asymmetric situation where a node A can reach node B, but B cannot reach A. Different transmit powers used at different nodes may also result in increased collisions, unless some precautions are taken. Suppose nodes A and B use lower power than nodes C and D. When A is transmitting a packet to B, this transmission may not be sensed by C and D. So, when C and D transmit to each other using a higher power, their transmissions will collide with the on-going transmission from A to B.

One simple solution (as a modification to IEEE 802.11) is to transmit request-to-send(RTS) and clear-to-send(CTS) at the highest possible power level but transmit DATA and ACK at the minimum power level necessary to communicate, as suggested in [10]. We refer to this as the BASIC scheme. Figure 1 illustrates the BASIC scheme. In Figure 1, nodes A and B send RTS and CTS, respectively, with the highest power level so that node C receives the CTS and defers its transmission. By using a lower power for DATA and ACK packets, nodes can conserve energy.

In the BASIC scheme, the RTS-CTS handshake is used to decide the transmission power for subsequent DATA and ACK packets. This can be done in two different ways as described below. Let p_{max} denote the maximum possible transmit power level. Suppose that node A wants to send a packet to node B. Node A transmits the RTS at power level p_{max}. When B receives the RTS from A with signal level p_r, B can calculate the minimum necessary transmission power level, $p_{desired}$, for the DATA packet based on received power level p_r, the transmitted power level, p_{max}, and noise level at the receiver B. If the procedure for estimating $p_{desired}$ is borrowed from [11], this procedure determines $p_{desired}$ taking into account the current noise level at node B. Node B then specifies $p_{desired}$ in its CTS to node A. After receiving CTS, node A sends DATA using power level $p_{desired}$. Since the signal-to-noise ratio at the receiver B is taken into consideration, this method can be accurate in estimating the appropriate transmit power level for DATA.

In the second alternative, when a destination node receives an RTS, it responds by sending a CTS as usual (at power level p_{max}). When the source node receives the CTS, it calculates $p_{desired}$ based on received power level, p_r, and transmitted power level (p_{max}), as

$$p_{desired} = \frac{p_{max}}{p_r} \times Rx_{thresh} \times c$$

where $R_{xthresh}$ is the minimum necessary received signal strength and c is a constant. A c is set equal to 1 in this simulations. Then, the source transmits DATA using a power level equal to $p_{desired}$. Similarly, the transmit power for the ACK transmission is determined when the destination receives the RTS. There are two assumptions. First, a signal attenuation between source and destination nodes is assumed to be the same in both directions. Second, noise level at the receiver is assumed to be below some predefined threshold. This alternative does not require any modification to the CTS format. This alternative is used in this simulation of BASIC and the proposed scheme.

3.2 Deficiency of the BASIC Protocol

In the BASIC scheme, RTS and CTS are sent using p_{max}, and DATA and ACK packets are sent using the minimum necessary power to reach the destination. When the neighbor nodes receive an RTS or

CTS, they set their NAVs for the duration of the DATA-ACK transmission. For example, in Figure 2, suppose node D wants to transmit a packet to node E. When D and E transmit the RTS and CTS respectively, B and C receive the RTS, and F and G receive the CTS, so these nodes will defer their transmissions for the duration of the D-E transmission. Node A is in the carrier sensing zone of D (when D transmits at p_{max}) so it will only sense the signals and cannot decode the packets correctly. Node A will set its NAV for EIFS duration when it senses the RTS transmission from D. Similarly, node H will set its NAV for EIFS duration following CTS transmission from E.

When transmit power control is not used, the carrier sensing zone is the same for RTS-CTS and DATA-ACK since all packets are sent using the same power level. However, in BASIC, when a source and destination pair decides to reduce the transmit power for DATA-ACK, the transmission range for DATA-ACK is smaller than that of RTS-CTS; similarly, the carrier sensing zone for DATA-ACK is also smaller than that of RTS-CTS. When D and E in Figure 2 reduce their transmit power for DATA and ACK transmissions respectively, both transmission range and carrier sensing zone are reduced. Thus, only C and F can correctly receive the DATA and ACK packets, respectively. Furthermore, since nodes A and H cannot sense the transmissions, they consider the channel to be idle. When any of these nodes (A or H) starts transmitting at the power level p_{max}, this transmission causes a collision with the ACK packet at D and DATA packet at E. This results in throughput degradation and higher energy consumption because of retransmissions.

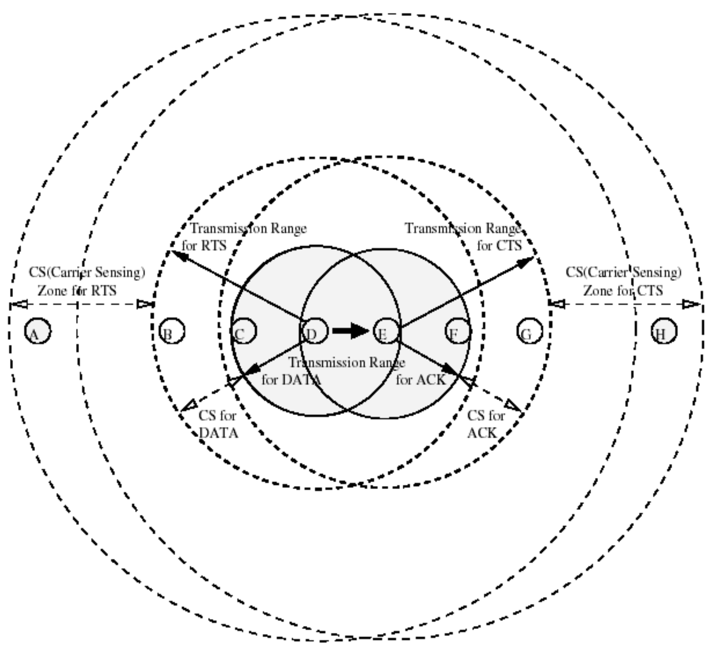


Figure 2. BASIC schemes : Transmission Range, Carrier Sensing Zone

3.3 Proposed Power Control MAC Protocol

Proposed Power Control MAC (PCM) is similar to the BASIC scheme in that it uses power level p_{max} for RTS-CTS and the minimum necessary transmit power for DATA-ACK transmissions. Here are descriptions of the procedure used in PCM.

1. Source and destination nodes transmit the RTS and CTS using p_{max}. Nodes in the carrier sensing zone set their NAVs for EIFS duration when they sense the signal and cannot decode it correctly.
2. The source node may transmit DATA using a lower power level, similar to the BASIC scheme.
3. To avoid a potential collision with the ACK (as discussed earlier), the source node transmits DATA at the power level p_{max}, periodically, for just enough time so that nodes in the carrier sensing zone can sense it.
4. The destination node transmits an ACK using the minimum required power to reach the source node, similar to the BASIC scheme.

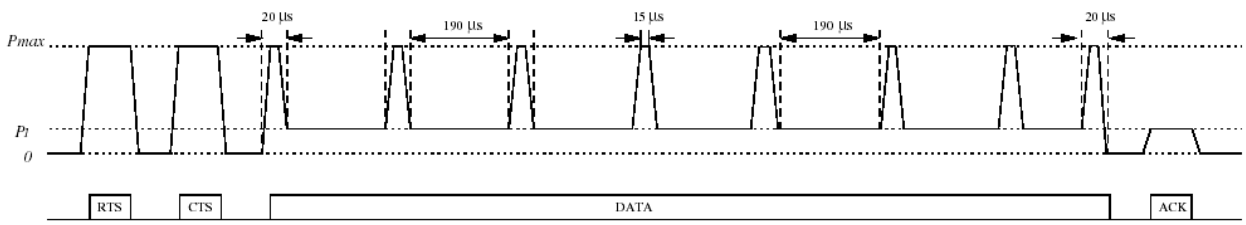
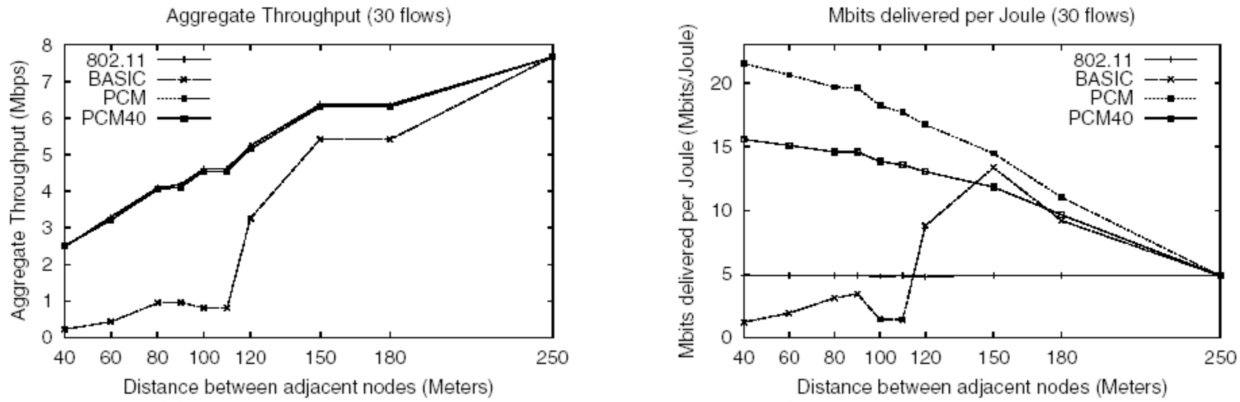


Figure 3. Signal Diagram for PCM

Figure 3 shows how the transmit power level changes during the sequence of an RTS-CTS-DATA-ACK transmission. After the RTS-CTS handshake using p_{max}, suppose the source and destination nodes decide to use power level p1 for DATA and ACK. Then, the source will transmit DATA using p1 and periodically use p_{max}. The destination uses p1 for ACK transmission. The key difference between PCM and the BASIC scheme is that PCM periodically increases the transmit power to p_{max} during the DATA packet transmission. With this change, nodes that can potentially interfere with the reception of ACK at the sender will periodically sense the channel as busy, and defer their own transmission. Since nodes that can sense a transmission but not decode it correctly only defer for EIFS duration, the transmit power for DATA is increased once every EIFS duration.

According to [12], 15 µs should be adequate for carrier sensing, and time required to increase output power ($power_{on}$) from 10% to 90% of maximum power (or power-down from 90% to 10% of maximum power) should be less than 2 µs. Thus, we believe 20 µs should be enough to power up (2 µs), sense the signal (15 µs), and power down (2 µs).

In this simulation, EIFS duration is set to 212 µs using a 2 Mbps bit rate. In PCM, a node transmits DATA at p_{max} every 190 µs for a 20 µs duration. Thus, the interval between the transmissions at p_{max} is 210 µs, which is shorter than EIFS duration. A source node starts transmitting DATA at p_{max} for 20 µs and reduces the transmit power to a power level adequate for the given transmission for 190 µs. Then, it repeats this process during DATA transmission. The node also transmits DATA at p_{max} for the last 20 µs of the transmission. With the above simple modification, PCM overcomes the problem of the BASIC scheme and can achieve throughput comparable to 802.11, but uses less energy. However, note that PCM, just like 802.11, does not prevent collisions completely. Specifically, collisions with DATA being received by the destination can occur, as discussed earlier. Our goal in this paper is to match the performance of 802.11 while reducing energy consumption. To be more conservative in estimating the energy consumption of PCM, we also perform our simulations where we increase the transmit power every 170 µs for 40 µs during DATA transmission. We refer to this variation as PCM40. This variation will consume more energy as compared to the above version of PCM.



(a) Aggregate throughput (b) Total data delivered per joule

Figure 4. Simulation Results

3.4. Simulation

Four different simulation condition were done, BASIC, PCM, PCM40, as well as 802.11. Two metrics are considered as follows.

•Aggregate throughput over all flows in the network.

•Total data delivered per unit of transmit energy consumption (or, Mbits delivered per joule).

3.4.1 Simulation Model and Topology

For simulations, ns-2 (ns-2.1b8a) is used with the CMU wireless extension. And 2 Mbps for the channel bit rate is used. Packet size is 512 bytes unless otherwise specified. Each flow in the network transmits CBR (Constant Bit Rate) traffic. The simulation with various network loads is performed. A mobility is not considered in this simulation. For network topologies, both a simple chain and random topologies are used. For the chain topology, 10 transmit power levels are considered, 1 mW, 2 mW, 3.45 mW, 4.8 mW, 7.25 mW, 10.6 mW, 15 mW, 36.6 mW, 75.8 mW, and 281.8 mW, which roughly correspond to the transmission ranges of 40 m, 60 m, 80 m, 90 m, 100m, 110 m, 120 m, 150 m, 180 m, and 250 m, respectively. For the random topology, four transmit power levels are considered, 2 mW, 15 mW, 75.8 mW, and 281.8 mW, roughly corresponding to the transmission ranges of 60 m, 120 m, 180 m, and 250 m, respectively.

Chain topology consists of 31 nodes with 30 single hop flows. The distance between adjacent node pairs is uniform. In these simulations, the distance is varied from 40 m to 250 m. For the random topology, 50 nodes are placed randomly within a 1000x1000 m^2 flat area. One flow originates at each node with the nearest node as its destination. Thus, a total of 50 flows are generated and 50 different random topologies (scenarios) are simulated.

3.4.2 Simulation Results

Only the result of chain topology is introduced here. The result of random topology is almost same as that of chain topology. Figure 4 shows the simulation results for 31 nodes with 30 flows in a chain topology. Each flow generates traffic at the rate of 1 Mbps. As the distance between two neighbors increases in Figure 4(a), the aggregate throughput of all schemes increases. This is because when nodes are far apart, a larger number of nodes can transmit simultaneously. PCM, PCM40 and 802.11 achieve comparable aggregate throughput as seen from the overlapping curves in Figure 4(a), but the BASIC scheme performs poorly in most cases. As the number of potential collisions becomes smaller, the aggregate throughput increases in Figure 4(a). The total data delivered per joule with the BASIC scheme is worse than 802.11 for many cases in Figure 4(b). This is due to poor aggregate throughput with BASIC and extra energy consumption from collisions and retransmissions. Since PCM40 consumes more energy compared to PCM, it gives less data delivered per joule, but it still performs better than 802.11, or BASIC (except for the 150 m distance). When the adjacent nodes are 250 meters apart, BASIC and PCM cannot reduce the transmit power for DATA-ACK. (Recall that the transmission range at p_{max} is 250 m.) Therefore, in Figure 4, all four schemes (802.11, BASIC, PCM and PCM40) perform the same when nodes are 250 m apart.

4. Bounded Slowdown Protocol [1]

4.1 PSM-static Impact on RTT

With PSM-static enabled, the network interface enters a sleep state whenever it is not sending or receiving data. When the mobile device has data to send (e.g., a TCP SYN or ACK packet, a TCP data packet containing a Web request, etc.), it can wake the network interface up at any time. However, the network interface will go to sleep as soon as this data has been transmitted to the AP. When the response data arrives from the server after some delay, it must be buffered at the AP until the next beacon occurs. This delay increases the observed RTT for the connection.

If the mobile device initiates a request/response transaction, the observed RTT depends on when it sends the request data relative to the beacon period. For example, with an actual RTT of 20 ms and a beacon period of 100 ms, if the mobile device sends the request immediately after a beacon, the response will be buffered at the AP and received after the next beacon; thus the observed RTT will be 100 ms. If the mobile device sends the request 79 ms after a beacon, the AP will receive the response just before the next beacon and the observed RTT will be just over 20 ms. However, if the mobile device sends the request 81 ms after a beacon, the AP will receive the response just after the next beacon and will have to buffer the data until the subsequent beacon; the observed RTT will be 120 ms, a factor of 6 slowdown.

In another test of PSM-static, the mobile client opens a TCP connection to a server and sends a request for some number of bytes; the server responds by sending the requested block of data. By doing this for power-of-two data transfer sizes between 1 Byte and 4 MBytes, we determined the relationship between data transfer size and transfer time. The client used was the same iPAQ device. The server was run on various machines to evaluate different network characteristics. The first server was in the same building and three network hops away from the AP; the RTT was 5 ms, and the bandwidth was at least 10 Mbps. The second server was located around 3000 miles and 20 network hops away and had a high bandwidth network path to the AP; the RTT was 80 ms and the bandwidth was at least 10 Mbps. The third server was located around 3 miles and 8 network hops away and behind a DSL network connection; it had a 50 ms RTT and outgoing bandwidth of 70 Kbps. Each performance test was run ten times alternating between PSM on and PSM off (five tests each). The results showed no significant variations between runs, and the mean values are presented.

Figure 5 shows the total transfer time (including the request and response) as a function of data transfer size for each server with both PSM on and PSM off. Figure 6 presents another view of the same data; it shows the slowdown incurred using PSM. For small data transfer sizes the entire response fits in one or two TCP data packets, and the total time for the transaction is equal to two RTTs - during the first RTT the client sends a SYN packet to the server, and the server responds with a SYN+ACK packet;

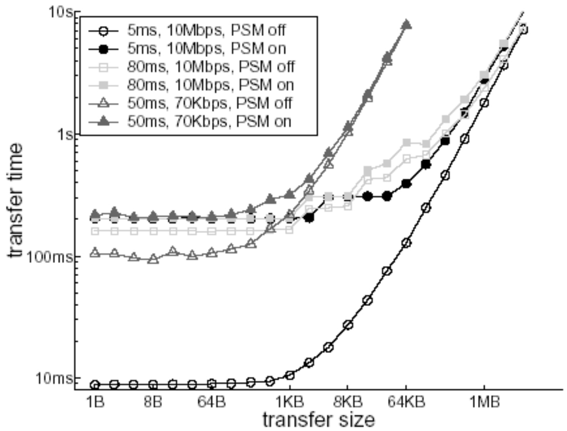


Figure 5. Data transfer size vs. transfer time

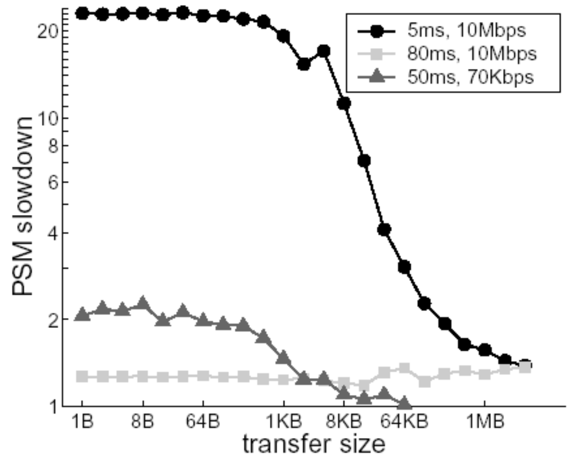


Figure 6. PSM slowdown

during the second RTT the client sends the request to the server and it responds with up to two data packets. With PSM off, the transfer time is determined by the RTT to each server; however, with PSM on, the transfer times are 200 ms regardless of the server.

4.2 Bounded-Slowdown (BSD) Protocol

This section presents the BSD protocol that employs an adaptive algorithm to maintain performance while minimizing the energy consumed by a wireless network interface. Its basic assumption is that, for request/response network traffic, the percentage increase in round trip times is more important than the absolute increase from the perspective of higher-layer protocols and human users. Formally, if the base RTT in the absence of PSM is R, then the goal is to minimize energy while limiting the observed RTT to $(1+p)\bullet R$; for a specific parameter $p > 0$, this limits the RTT increase to $100\bullet p$ percent. We present an optimal algorithm that meets this goal. We start with an observation about sleep durations:

LEMMA 1. *If, after sending a request at time $t_{request}$ the mobile device has received no response at time $t_{current}$, then the network interface can go to sleep for a duration up to $(t_{current} - t_{request})\bullet p$ while bounding the RTT slowdown to $(1+p)$*

This is true because for the greatest slowdown, the actual RTT, $R_{actual} = (t_{current} - t_{request})$, and the observed RTT, $R_{observed} = R_{actual} + R_{actual}\bullet p$; therefore $R_{observed} \leq (1+p)\bullet R_{actual}$.

To minimize energy, an optimal algorithm must clearly always put the network interface into the sleep state as soon as possible and for as long as possible. However, to bound the slowdown, the mobile device must periodically check with the AP for buffered data as governed by Lemma 1. Therefore, if (for the moment) we neglect synchronization constraints between the wireless network interface and the AP, we can state the following theorem:

THEOREM 1. *To minimize energy while bounding RTT slowdown to a factor $(1+p)$, a network interface should go to sleep an infinitesimally short period of time after it sends any request data, and only wake up to check for response data as governed by Lemma 1.*

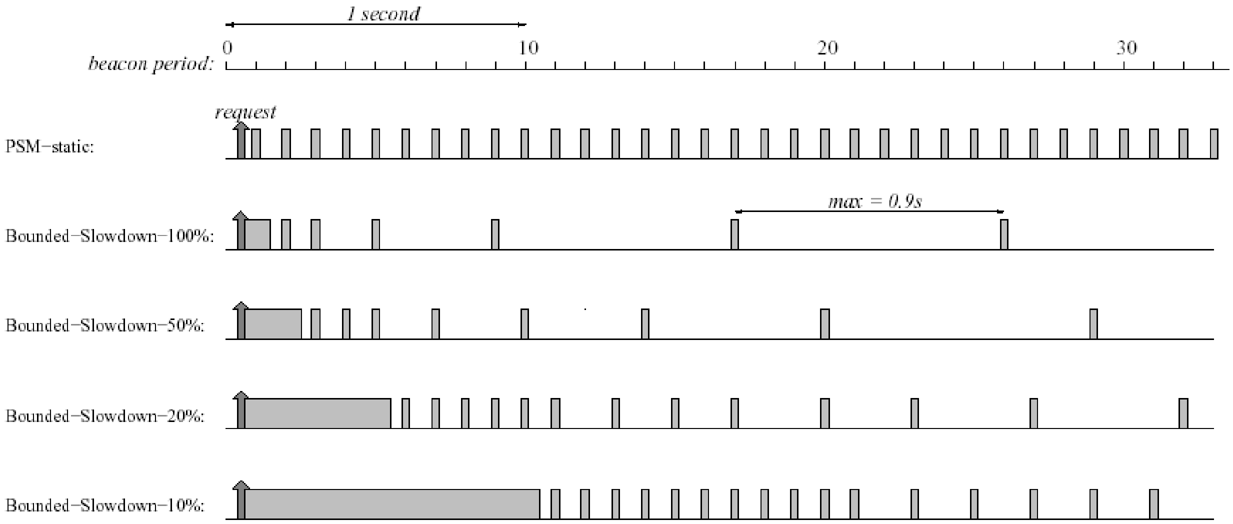


Figure 7. The arrow indicates a request by the mobile device, the initial shaded area indicates when BSD stays awake for a set time T_{awake} after the request, and the shaded bars indicate when the network interface wakes up to listen to beacons.

Figure 7 shows the behavior of PSM-static and the BSD protocol for various values of *p* (these are labeled as 100•*p* percent). To allow direct comparisons with the 802.11 PSM, we set T_{bp} (Fixed beacon period) to 100 ms. Additionally, in the implementation the BSD protocol sets the maximum sleep duration to 0.9 s to avoid TCP timeouts.3 Considering one example in Figure 7, when p=0.2 (20%), $T_{awake} = T_{bp}/p$ = 500ms, so the network interface stays awake for half a second after the mobile device sends a request. Then, it begins sleeping and waking up to listen to every beacon while T_{sleep} is rounded down to 100 ms. After a second has elapsed since the request, T_{sleep} is 200 ms, so it sleeps for two beacon periods, and so on.

In summary, with the BSD protocol, fast response times are not delayed, while longer ones are increased by up to a parametrized maximum factor, 1+p. Compared to PSM-static, the active energy is increased since the transition to sleep mode is delayed, but the energy spent listening to beacons is decreased due to the longer sleep intervals.

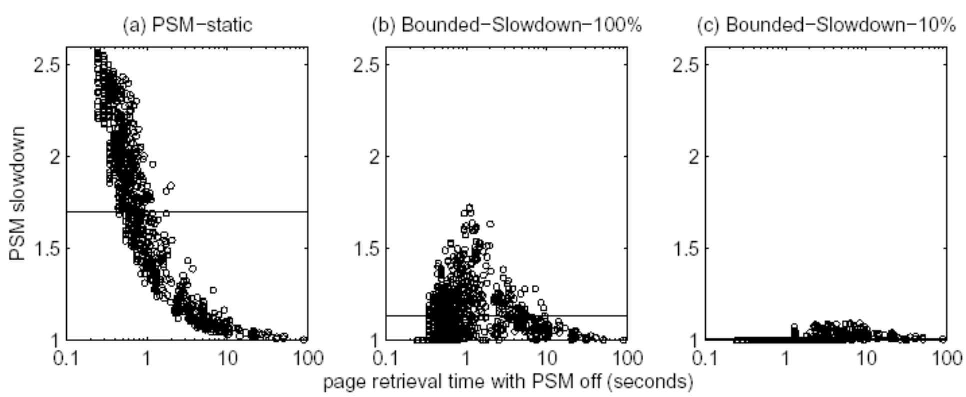


Figure 8. Per-page slowdown. Each marker represents a single Web page. Mean slowdown of each graph is 1.69, 1.14, 1.01 respectively

4.3 Simulation methodology and result of web page retrieval

The simulation is done by using ns-2 and the model is mobile client communicating with an access point over a wireless link with PSM. Sleep mode is simulated by deactivating the queue elements of a link so that they do not forward any packets, and waking up simply entails activating the queues. The beaconing is implemented using a timer that expires every 100ms. To evaluate PSM-static and BSD, a network consisting of a mobile client, an access point, and a server are modeled. To simulate the power consumption of the 802.11 network interface card, they modeled the power usage as 750 mW while awake, and 50 mW while asleep.

Figure 8 shows that PSM-static has the greatest negative impact on pages with fast retrieval times. These are slowed down by up to about 2.5 times which is the penalty for extending a 40 ms RTT to 100 ms. BSD-100% shows a large improvement, and does bound the worst-case slowdown to be smaller than 2 times. In fact, all of the slowdowns are far less than this bound because the protocol keeps the network interface awake for 100 ms after the mobile device sends data, so fast RTTs are not slowed down at all. BSD-10% further improves performance and shows almost no slowdown.

Enabling PSM-static reduces energy by about a factor of 11, but suffers from a slowdown of 16-232% depending on the server RTT. Based on the estimates, the energy spent while awake is negligible since the network interface is in sleep mode for around 1000 times longer than it is awake. Waking to listen to beacons accounts for 23% of the total power consumption;

To improve performance as the slowdown parameter is decreased, BSD successively increases the awake energy since it stays awake for longer after the mobile device sends data. The awake energy also increases with slower server RTTs since BSD typically remains awake for entire TCP data transfers, and these become longer. However, BSD also reduces the energy spent listening to beacons since it adaptively increases the listen interval when there is no activity. The listen energy is reduced by 6.8 times with BSD-10% and 8.2 times with BSD-100%, close to the maximum reduction of 9 times that would be achieved by listening every 900 ms (the maximum listen interval we allow) instead of every 100 ms. Combining these two energy effects, BSD uses even less energy overall than PSM-static in many cases, and even in the worst case it only increases the energy by 26%.

5. Conclusions

We have investigated two different protocols for power saving algorithms of mobile device in wireless networks. Each one is derived from the different motivations, one is for ad hoc network and based on IEEE 802.11 MAC protocol. When the BASIC scheme is used, nodes in the carrier sensing zone of RTS-CTS can cause collisions with on-going DATA-ACK transmissions because these nodes may not sense DATA transmission which may use a lower transmit power. These can result in more collisions, more energy consumption, and throughput degradation. But PCM periodically increase the transmit power during DATA transmission, which can achieves energy savings without causing throughput degradation.

The other is for infrastructure network and based on IEEE 802.11 PSM-static protocol. PSM-static protocol can reduces the energy consumed during Web access by 11 times compared to no PSM. But it increases the average Web page retrieval time by 16-232%. To overcome these problems, BSD protocol is presented which adapts to network activity dynamically. The BSD stays awake for a short period of time after a request is sent, and listens to fewer beacons when the link remains idle. Compared to PSM-static, BSD reduces average Web page retrieval times by 5-64% and reduces energy consumption by 1-14%.

6. Discussions

Both of the energy saving protocols are simulated by using ns-2. To prove the actual saving power of PSM, the experiment is done by using Enterasys Networks RoadAbout NIC. Concerned about PCM, it requires a frequent power increase and decrease of transmission which may make the implementaion difficult. It is not verified whether the energy is consumed more or not when the transmitting power is up and down.

References

[1] R.Krashinsky, H. Balakrishnan. Minimizing energy for wireless web access with bounded slowdown. ACM MOBICOM, pages 119-130, Atlanta, GA, September 2002

[2] E. Jung, N. H. Vaidya, A power control MAC protocol for Ad Hoc networks. ACM MOBICOM, pages 36-47, Atlanta, GA, September 2002.

[3] C. E. Jones, K. M. Sivalingam, P. Agrawal, J. C. Chen, A survey of energy efficient protocols for wireless networks, pages 343-358, Wireless Networks, 2001

[4] J. Macker and M. Corson, Mobile ad-hoc networking and the IETF, ACM Mobile Computing and Communications Review 2(1), 1998.

[5] M. Stemm and R.H. Katz, Measuring and reducing energy consumption of network interfaces in hand-held devices, IEICE Transactions on Fundamentals of Electronics, Communications, and Computer Science, 1997.

[6] S. Udani and J. Smith, Power management in mobile computing (a survey), University of Pennsylvania (1996), http://www.cis.upenn.edu/~udani/papers.html. and at AT&T Labs in Whippany, NJ.

[7] A. Chandrakasan and R.W. Brodersen, Low power digital CMOS design (Kluwer Academic, Norwell, MA, 1995).

[8] B. Chen, K. Jamieson, H. Balakrishnan, R. Morris. Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. ACM Wireless Networks, 8(5), September 2002.

[9] IEEE Computer Society LAN MAN Standards Committee. IEEE Std 802.11: Wireless LAN medium access control and physical layer specifications, August 1999.

[10] S. Agarwal, S. Krishnamurthy, R. H. Katz, and S. K. Dao. Distributed power control in Ad-hoc wireless networks. PIMRC01, 2001.

[11] J. P. Monks, V. Bharghavan, W. mei W. Hwu. A power controlled multiple access protocol for wireless packet networks. In INFOCOM 2001, April 2001.

[12] The editors of IEEE 802.11. Wireless LAN medium access control (MAC) and physical layer (PHY) Specification, 1997.