Energy Efficient Design of Portable Wireless Systems

Tajana Simunic Haris Vikalo* Peter Glynn† Giovanni De Micheli

Computer Systems Laboratory
Stanford University

*Information Systems Laboratory
Stanford University

†Management Science and Engineering Department
Stanford University

ABSTRACT

Portable wireless systems require long battery lifetime while still delivering high performance. The major contribution of this work is combining new *power management* (PM) and *power control* (PC) algorithms to trade off performance for power consumption at the system level in portable devices. First we present the formulation for the solution of the PM policy optimization based on renewal theory. Next we present the formulation for power control (PC) of the wireless link that enables us to obtain further energy savings when the system is active. Finally, we discuss the measurements obtained for a set of PM and PC algorithms implemented for the WLAN card on a laptop. The PM policy we developed based on our renewal model consumes three times less power as compared to the default PM policy for the WLAN card with still high performance. Power control saves additional 53% in energy at same bit error rate. With both power control and power management algorithms in place, we observe on average a factor of six in power savings.

1. INTRODUCTION

Battery-operated portable systems demand tight constraints on energy consumption. Better low-power circuit design techniques and advances in battery technology have helped to increase battery lifetime. On the other hand, managing power dissipation at higher levels can considerably reduce energy consumption, and thus increase battery lifetime [1].

In this paper we consider a client-server system, where the client is a portable device capable of wireless digital communication, such as a laptop. Each communication "session" is initiated by the client (e.g. web browsing or telnet). Our main contribution is combining new power management and power control algorithms to reduce power consumption of the wireless card. Power management reduces power by selectively placing the card into states with lower power consumption when the user is not actively communicating via the card. Power control reduces the level at which the card transmits while keeping the same performance. Thus our work reduces power consumption in both active and idle modes of card's operation.

As soon as a wireless card is added to a portable system, the energy consumption can be drastically increased. For example, a *wireless local area network* (WLAN) card with 1Mb/s rate takes up to 12 times more power than 10Mb/s Ethernet card [2]. Wireless cards have two active states - receive (listening for data) and transmit (sending data out). The card we used has also two low-power states: doze and off. Doze mode consumes much less power than the active states and can be entered and exited with very small performance penalty (e.g. 0.8ms). When the card is turned off, it consumes no power, but takes longer to transition into active state (e.g. 60ms).

The new IEEE 802.11 standard defines power management for wireless LAN at *medium access control* (MAC) and physical layers [3]. The standard requires that a central *access point* (AP) send out a beacon every 100ms followed by a *traffic indication map* (TIM). Each card that desires to communicate listens for the beacon in order to synchronize the clock with AP, and for the TIM to find out if any data is arriving for it. If it does not need to transmit or receive, the card can then go to doze state until the next beacon.

The IEEE standard does not address the need for power management at the system level. If the card is turned off when it is not being used, much larger power savings can be observed. We define a *power manager* (PM) that observes the workload of the system and decides when and how to force the power state transition according to the *power management policy.*

A heuristic power management policy implemented in most operating systems is a *timeout policy* that tends to waste power while waiting for the timeout to expire. Heuristic predictive policies developed for interactive terminals [4, 5] force the transition to a low power state as soon as a component becomes idle if the predictor estimates that the idle period will last long enough. A wrong estimate can cause both performance and energy penalties.

In this paper we introduce a new model for power management at the system level that guarantees optimal results. After collecting and analyzing a set of traces for web browsing and telnet application, we found that the user request arrivals can be modeled with Pareto distribution. The card transitions between off and active states are best fit with uniform distribution. Our model has two non-exponential transitions occurring at the same time when the card transitions from doze mode into off state. Thus we could not apply policy optimization algorithms based on exponential models, such as [9, 10, 13]. Large errors result if exponential distribution is used for all transitions, as was shown in [11]. Another approach to handle non-exponential transitions is to use adaptive method, such as in [12]. This method requres policy interpolation at very short time increments, regardless of the device state, thus causing an increase in CPU and memory energy consumption. In order to correctly model the system, we formulated the policy optimization problem using renewal theory [16]. The optimal policy we obtain has been implemented on a laptop for the WLAN card. Our results show that reduction in power can be as large as a factor of five with a small performance penalty.

In addition, we implemented a power control algorithm that reduces power consumption of the wireless card during transmission. Power control for minimizing energy consumption subject to maintaining a required transmission rate is considered in, e.g., [19]. There, authors discuss solutions to various constant transmitter power and constant *signal-to-interference ratio* (SIR) problems while assuming normalized wireless link so that transmitted and received power are equal. Each mobile is deciding on its own optimal transmitting power, hence the proposed power control algorithms are of distributed nature. Zorzi et al. [20] studied a related problem of error control in an energy-constrained wireless network, optimizing transmission power and transmission strategy for maximum throughput. In the current paper, we implement an algorithm that achieves the required QoS by maintaining constant target SIR at the AP. The algorithm is centralized since the AP decides the power level of each user. The transmitter power is a function of the target SIR and time-varying parameters of the channel. The WLAN card we used in our experiments employs *direct sequence code division multiple access* (DS-CDMA) scheme. The capacity of DS-CDMA is interference limited. By maintaining the transmission power of each user's card on the minimum level needed to achieve required *bit-error rate* (BER), we hold the interference to the smallest tolerable amount while at the same time reducing client's power consumption. Our power control algorithm saves as much as 53% of the power as compared to the default algorithm implemented in the card we were using.

Section 2 develops the model for power management based on renewal theory. Theoretical background for power control is presented in Section 3. We discuss measurement results of a wireless card in Section 4. Finally, we summarize our findings in Section 5.

2. POWER MANAGEMENT

We optimize energy consumption under performance constraint (or vice versa) to obtain an optimal power management policy. The system we model consists of the user, the wireless card and the queue (the buffer associated with the card). In the next sections we describe the system model we developed, followed by a new approach based on renewal theory to obtain an optimal power management policy.

2.1 Wireless Card Model

The wireless card has multiple power states: two active states, *transmitting, receiving*, and two inactive states, *doze* and *off*. Transmission power is 1.65W, receiving 1.4W, and the power consumption in the doze state is 0.045W [7] and in off state it is 0W. When the card is awake (not in the off state), every 100ms it synchronizes its clock to the access point (AP) by listening to AP beacon. After that it listens to TIM map to see if it can receive or transmit during that interval. Once both receiving and transmission are done, it goes to doze state until the next beacon. This portion of the system is fully controlled from the hardware and thus is not accessible to the power manager that has been implemented at the OS level.

The power manager can control the transitions between the doze and the off states. Once in the off state, the card waits for the first user request arrival before returning back to the doze state. We measured the transitions between doze and off states using *cardmgr* utility in Linux system. The transition from the doze state into the off state takes on average $t_{ave} = 62ms$ with variance of $t_{var} = 31ms$. The transition back takes $t_{ave} = 34ms$ with $t_{var} = 21ms$ variance. The transition between doze and off states are best described using uniform distribution, where t_0 and t_1 can be defined as $t_{ave} - t_{var}$ and $t_{ave} + t_{var}$ respectively. The cumulative probability function for the uniform distribution is shown below.

$$F_{wlan}(t) = \begin{cases} \frac{t-t_0}{t_1-t_0} & t_0 \leq t \leq t_1 \\ 0 & otherwise \end{cases} \tag{1}$$

The wireless driver uses a buffer for storing requests that are not currently being serviced during busy times and transition times. We model the buffer using queue. We characterize card's power states by the number of jobs pending for service in the queue.

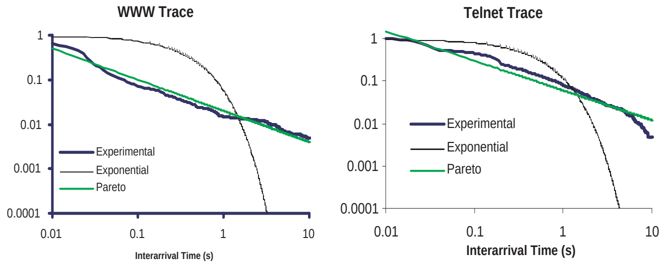

Figure 1: Tail distribution of request arrivals

2.2 User Model

We analyzed user's behavior in accessing the wireless card on a laptop running Linux with telnet and web browser using *tcpdump* utility [8]. Figure 1 shows the tail distributions of measured results fitted with Pareto and exponential distributions. Pareto distribution shows a much better fit for the arrival after a longer idle time as compared to the exponential distribution. Exponential distribution (or Poisson arrivals) can be used to model subsequent short request interarrival times. Pareto cumulative distribution is defined in Equation 2 with parameters $a = 0.7$ and $b = 0.02$ for web requests and $a = 0.7$ and $b = 0.06$ for telnet requests:

$$F_{user}(t) = 1 - at^{-b} \tag{2}$$

2.3 Renewal theory model

The goal of power management optimization is to minimize performance penalty under energy consumption constraint (or vice versa). In this paper we focus on the former problem. The system model is shown in Figure 2, and the system states are shown in Figure 3.

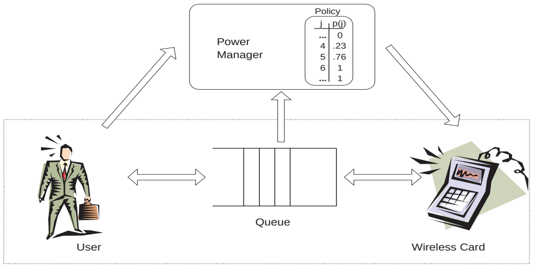

Figure 2: System Model

In our system, the only decisions PM can make is whether to leave the card in doze state or turn it off completely. If the card is left on, it will wait in the doze state until the first request arrives from the user. Upon arrival of request, the card transitions into active state. If the PM turns off the card, the system enters a transition state between doze and off state. The transition state models the fact that it takes a finite amount of time to actually turn off the card as modeled by Equation 1. If during the transition time a request arrives from the user, the card will enter the transition to active state as soon as transition to off state is completed. If no request arrives during the transition to off state, the card stays off until the next request arrives. Upon request arrival, the card starts the transition back into active state. This transition also takes a finite amount of time, and thus is modeled as a transition state. Once transition into active state is completed, the card either transmits or receives, and then again returns to doze state.

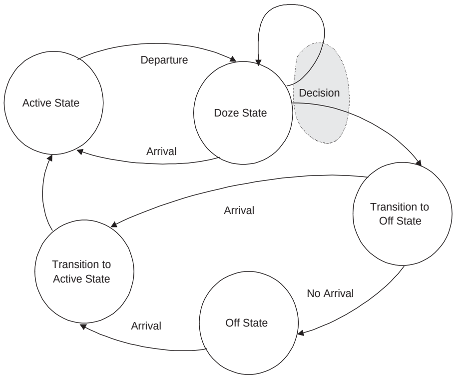

Figure 3: System states

Renewal theory [16, 17] describes counting processes for which the request interarrival times are independent and identically distributed with arbitrary distributions. A process can be considered to be renewal process only if there is a state of the process in which the whole system is probabilistically restarted. The complete cycle of transition from doze state, through other states and then back into doze state can be viewed as a renewal period [17]. Every time the system reenters doze state, the time is measured again from zero. Renewal theory can handle multiple non-exponentially distributed transitions occurring at the same time. In particular, we can model changes of queue state due to user request arrivals (Pareto distribution) which occur during the transition from doze into off state or from off to active state (uniform distribution). It can be shown [17] that with renewal theory the search for the best policy for a system modeled using stationary non-exponential distributions can be cast into a stochastic control problem whose solution is randomized in the presence of constraints. The randomization is a function of time elapsed from the last entry into the reset state.

In our case, the problem of power management policy optimization is to determine the optimal distribution of random variable Γ that specifies when the transition from the doze state to off state should occur from the last renewal time of the system (the last entry into doze state). We assume that Γ takes on discrete values in $[0, h, 2h, ...jh, ...)$, where j is an index of time from reset state and h is a fraction of the time it takes for a system to transition from doze to off state. With smaller values of h the solution gets more accurate, the computation time grows as well. The solution to policy optimization can be viewed as a table of probabilities for transition from doze to off state, $p(j)$, indexed by time values jh. Furthermore, when user request arrival times are modeled using Pareto distribution, it can be shown that the optimal policy always has the probability of turning the card off monothonically increasing with time [17]. The policy table thus has entries $p(j)$ that increase with time from zero until some time kh where the probability of giving a command to turn off the card is equal to one. Thus we can simplify the implementation of PM. If the PM's decision upon entry to doze state is to leave the card on, the power manager waits until the time jh where the probability of transition to off state, $p(j)$, is greater than zero. From that point on, the decision is reevaluated every h seconds until either the card is turned off or a user request arrives and the card transitions into the active state.

Based on renewal theory we formulate optimization problem to minimize the average number of jobs waiting in the queue (performance penalty) under power constraint:

$$\begin{aligned}
\textbf{LP:}\ \min \quad & \sum_j q(j)x(j) \\
\text{s.t.} \quad & \sum_j [e(j) - t(j)Constraint]x(j) = 0 \\
& \sum_j x(j) = 1/E[\tau] \\
& \sum_j t(j)x(j) = 1
\end{aligned} \tag{3}$$

The number of job requests in the queue, $q(j)$, is derived using:

$$q(j) = E[\int_0^\tau Q(t)dt] = \sum_j p(j)E[\int_0^\tau Q(t)dt|\Gamma = jh] \qquad (4)$$

where $Q(t)$ is the number of jobs in the queue, $p(j)$ is the probability of issuing a command to go to doze state at time jh and $\Gamma = jh$ signifies that the command to turn the card off is given at time jh. The expected time until renewal given that the system went to off state at time jh, is given by:

$$t(j) = E[\tau_j|\Gamma = jh] \qquad (5)$$

The expected energy consumption can be defined by:

$$e(j) = E[\int_0^\tau W(t)dt] = \sum_j p(j)E[\int_0^\tau W(t)dt|\Gamma = jh] \qquad (6)$$

where $W(t)$ is the expected power consumption for each state of the device during the renewal time.

The LP solves for the probability frequencies, $x(j)$. The probability of transition to off state from doze state at time jh, $p(j)$, can be computed using expected time until renewal, $E[\tau] = \sum_j p(j)E[\tau_j|\Gamma = jh]$:

$$p(j) = x(j)E[\tau] \qquad (7)$$

The optimal policy is a table of probabilities $p(j)$. PM observes the state of the system and user request arrivals at run time. Once the system enters doze state, PM starts keeping track of elapsed time. The decision to turn off the card is first evaluated at the time jh for which the table gives non-zero probability $p(j)$ of turning the card off. For each $p(j)$ that is not zero, PM generates a random number using a pseudo-random number generator. If the number generated is less than $p(j)$, PM shuts down the card. Otherwise the card stays in the doze state until either the next time probability of shutting down the card is non-zero, or until request arrival that forces the card to transition into active state. Once the card is turned off, it stays off until the first request arrives, at which point it transitions into active state.

In this section we presented a new algorithm for power management of the wireless card that saves energy by selectively turning off the card. In the next section we present a power control algorithm that saves energy by reducing the transmission power level when the card is active.

3. POWER CONTROL

In this section, we discuss implementation of uplink power control in the wireless system. Figure 4 shows the system configuration we focus on for power control. On every transmission from each client, *access point* (AP) measures received *signal to interference and noise ratio* (SIR) and *bit error rate* (BER). AP then performs quick calculation using the closed form optimal solution. The result of calculation is the optimal transmission level for each of the clients in the system. Upon receiving a response from AP, each client adjust the transmission power level accordingly. The power adjustement can be done very rapidly as compared to the time needed to set up and send data, and thus does not need to be considered as overhead in this process.

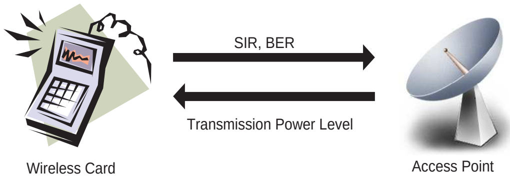

Figure 4: Power control system

In DS-CDMA scheme, data sequence to be transmitted is first modulated using a narrowband modulation scheme and then spread by multiplying with the spreading sequence, where each user is assigned a unique spreading sequence. This wideband signal is then transmitted over a wireless channel. We assume a commonly used *wide-sense stationary uncorrelated scattering channel* (WSSUS) model. The fading is assumed to be slow in comparison with the symbol rate so that channel remains constant over each symbol transmission period.

The mobile user specifies a minimum tolerable reliability in terms of bit error rate (BER) which can be mapped into required SIR [21]. The goal of power control is to minimize the total transmitted power subject to the constraint that the SIR requirement, $SIR_{req,i}$ for each user i is satisfied, i.e.,

$$\min\ p(t) = \sum_{i=1}^{M} \gamma_i(t)p_i(t) \qquad (8)$$

subject to:

$$\frac{G_i a_i(t)p_i(t)}{\sum_{j\neq i} a_j(t)\gamma_j(t)p_j(t) + \sigma(t)^2} \geq \frac{SIR_{req,i}}{1 - SIR_{req,i}\frac{M}{G_i}},$$

$$p_i(t) \geq 0, \ \text{for} \quad i = 1, 2, \ldots M,$$

where $p_i(t)$ is power of the ith user, γ_i is an indicator function ($\gamma_i = 1$ if ith user is currently active and $\gamma_i = 0$ otherwise), G_i is the processing gain of the ith user, $a_i(t)$ is the path loss form the ith user to the access point, M is the number of users in the network and $\sigma^2(t)$ is the noise power.

Optimization problem (8) can be rearranged as a linear program which has a solution iff (see [22])

$$\sum_{i=1}^{M} \gamma_i(t)\alpha_i < 1,$$

where

$$\alpha_i = \frac{SIR_{req,i}}{(1-M)SIR_{req,i} + G_i},$$

which is satisfied under the assumptions that we made. Moreover, the linear program can be solved in closed form (see [22]). Thus, the optimal transmission power $p_i(t)$ for each active card i at time t is calculated by the access point using:

$$p_i(t) = \frac{\gamma_i(t)\alpha_i}{a_i(t)} \frac{\sigma^2(t)}{1 - \sum_{j=1}^{M} \gamma_j(t)\alpha_j}. \qquad (9)$$

From the requirements on the transmission quality (i.e., $SIR_{req,i}$), the access point calculates α_i for each client in the network. When client i makes a request for a transmission, $\gamma_i(t) = 1$. The access point then takes the corresponding α_i, along with the measured values of $\sigma^2(t)$ and $a_i(t)$, and evaluates the ith client's optimal transmission power $p_i(t)$ from Equation 9. This value is then transmitted to the i^{th} client, whose card adjusts its transmission power accordingly. The optimal transmission power $p_i(t)$ takes on continuous values. The card can adjust its power only in discrete steps. Hence $p_i(t)$ is discretized to the closest supported value of the transmission power.

In the previous sections we described theoretical backgrounds for power management and power control algorithms. In the next section we present results that show large power savings we can obtain with both algorithms.

4. RESULTS

In this section we present the measurement results for power management algorithm we developed, followed by the power control algorithm results. Finally we show that the two approaches combined can give on average a factor of six in power savings, while keeping constant bit error rate. For all our measurements we used Lucent's WLAN 2Mb/s card [7] running on the laptop with Linux OS. As mobile environment is continually changing, it is not possible to reliably repeat the same experiment. As a result, we needed to use trace-based methodology discussed in [14].

The methodology consists of three phases: collection, distillation and modulation. We used *tcpdump* [8] utility to get the user's trace for two different applications: web browsing and telnet. At the same time, we collected card's transmission speed and SIR using a utility we developed that interfaces with the driver. We modeled transmission power levels of the card using measured SIR values, as we did not have access to power control functions of the WLAN card due to proprietary reasons. During distillation we prepared the trace for the modeling step. We had a LAN-attached host read the distilled trace and delay or drop packets according to the parameters we obtained from the measurements. In this way we were able to recreate the experimental environment, so that different algorithms can be reliably compared.

We perform power management policy optimization for our policy using *lp_solve* [18]. The optimization runs in about 30 seconds on a 300MHz Pentium processor. We implemented three different versions of our policy for each application with different power and performance penalty values (Renewal a,b,c for web browser and Renewal 1,2,3 for telnet). Since web and telnet arrivals behave differently (see Figure 1), we observe through OS what application is currently actively sending and use the appropriate power management policy.

In addition to measuring the energy consumption (and then calculating average power), we also quantified performance penalty using three different measures. Delay penalty, T_p, is the time system had to wait to service a request since the card was in the sleep state when it should not have been. In addition, we measure the number of shutdowns, N_{sd} and the number of wrong shutdowns, N_{wd}. A shutdown is viewed as wrong when the sleep time is not long enough to make up for the energy lost during transition between the doze and off state. The number of shutdowns is a measure of how eager the policy is, while a number of wrong shutdowns tells us how accurate the policy is in predicting a good time to shut down the card.

Table 1: Power Management for Wireless Web Browser

Policy	N_{sd}	N_{wd}	$T_{penalty}$ (sec)	P_{ave} (W)
Oracle	395	0	0	0.467
Renewal(a)	363	96	6.90	0.474
Renewal(b)	267	14	1.43	0.477
Competitive	623	296	23.8	0.479
Renewal(c)	219	9	0.80	0.485
Poisson	3424	2866	253.7	0.539
Default	0	0	0	1.410

Table 2: Power Management for Telnet Application

Policy	N_{sd}	N_{wd}	$T_{penalty}$ (sec)	P_{ave} (W)
Oracle	766	0	0	0.220
Renewal(1)	798	21	2.75	0.269
Renewal(2)	782	33	2.91	0.296
Competitive	780	40	3.81	0.302
Renewal(3)	778	38	3.80	0.310
Poisson	943	233	20.53	0.361
Default	0	0	0	1.410

The measurement results for a 2.5hr web browsing trace are shown in Table 1. Our algorithms (Renewal a,b,c) show on average a factor of three in power savings with low performance penalty. The competitive algorithm [15] guarantees to be within a factor of two of oracle policy. Although its power consumption is low, it has a performance penalty that is an order of magnitude larger than for our policy. A policy that assumes Poisson arrivals only [13] has a very large performance penalty because it makes the decision as soon as the system enters doze state.

Table 2 shows the measurement results for a 2hr telnet trace. Again our policy performs best, with a factor of five in power savings and a small performance penalty. Telnet application allows larger power savings because on average it transmits and receives much less data then the web browser, thus giving us more chances to shut down the card.

Table 3: Power control savings

BER	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Power Savings	53%	40%	32%	27.5%

We have tested our power control algorithm on an extensive set of trace-based recorded channel information. All the measurements were done in a building with a single access point (AP). When AP receives request for the transmission from a user, it calculates the optimal transmission power. AP then sends the command to user's wireless card via control channel to set the next transmission power to the calculated level. Measurements presented in [6] show that when no power management policy for turning the card off is implemented, the WLAN card by default spends about 5% of time in doze mode and the rest of the time in one of the active states (transmitting or receiving). Table 3 lists power savings obtained with respect to the power consumption when no power control algorithm is implemented. Clearly, higher requirements on the transmission quality (i.e., lower BER) yield higher power consumption.

Table 4: Power management and power control savings

`Application`	`PM Algorithm`	`BER`	P_{ON}/P_{PMPC}
`WWW`	`Renewal (a)`	10^{-3}	`6.3`
		10^{-6}	`4.1`
`Telnet`	`Renewal (1)`	10^{-3}	`9.0`
		10^{-6}	`6.7`

Finally, we combined power management with power control. The results are shown in Table 4 for both web browser and telnet applications. We report results for two BERs that are commonly used in practice. P_{ON} is power consumption of the original card design. P_{PMPC} is card's power consumption when both power management and power control algorithms are active. Thus, P_{ON}/P_{PMPC} gives us a factor of power savings. As power management saves energy by shutting down the card when it is not in use, and power control saves power by lowering transmission level in the active state, we are able to see large savings. Again, telnet application offers higher savings due to its lower bandwidth requirements.

5. CONCLUSIONS

We have presented a combination of two approaches aimed at saving power for wireless applications. We first discussed a new approach based on renewal theory for optimizing power management policies of wireless portable systems. The measurements show that our policy gives on average three times lower power consumption for the wireless card as compared to the current card implementation.

In addition, we presented an implementation of a power control algorithm for the wireless card with power savings as high as 53%. The combination of power management and power control algorithm presented in this work give on average a factor of six in power savings.

6. ACKNOWLEDGMENTS

This work was supported in part by NSF under grant number CCR-9901190. In addition, we would like to thank Hewlett-Packard Laboratories and Lucent Corporation for their support.

7. REFERENCES

[1] C. Ellis, “The case for higher-level power management”, *7th IEEE Workshop on Hot Topics in Operating Systems*, pp.162–167, 1999.

[2] D. Newman, K. Tolly, “Wireless LANs: How Far? How Fast?”, *Data Communication*, March, 1995.

[3] The Editors of IEEE 802.11, *IEEE P802.11D5.0 Draft Standard for Wireless LAN*, July, 1996.

[4] M. Srivastava, A. Chandrakasan. R. Brodersen, “Predictive system shutdown and other architectural techniques for energy efficient programmable computation,” *IEEE Transactions on VLSI Systems*, vol. 4, no. 1, pp. 42–55, March 1996.

[5] C.-H. Hwang and A. Wu, “A Predictive System Shutdown Method for Energy Saving of Event-Driven Computation”, in *Proceedings of the Int.l Conference on Computer Aided Design*, pp. 28-32, 1997.

[6] H. Woesner, J. Ebert, M. Schlager, A. Wolisz, “Power-Saving Mechanisms in Emerging Standards for Wireless LANs: The MAC Level Perspective”, *IEEE Personal Communications*, pp.40–48, June, 1998.

[7] Lucent, *IEEE 802.11 WaveLAN PC Card - User's Guide*, p.A-1.

[8] V. Jacobson, C. Leres, S. McCanne, *The Tcpdump Mannual Page*, Lawrence Berkeley Laboratory.

[9] L. Benini, A. Bogliolo, G. Paleologo and G. De Micheli, “Policy Optimization for Dynamic Power Management”, in *Transactions on CAD*, 1999.

[10] T. Simunic, L. Benini and G. De Micheli, “Event-driven power management”, *Proceedings of ISSS*, 1999.

[11] T. Simunic, L. Benini and G. De Micheli, “Power Management of Laptop Hard Disk”, *Proceedings of DATE*, 2000.

[12] E. Chung, L. Benini and G. De Micheli, “Dynamic Power Management for non-stationary service requests”, *Proceedings of DATE*, pp. 77–81, 1999.

[13] Q. Qiu, Q. Wu and M. Pedram, “Stochastic Modeling of a Power-Managed System: Construction and Optimization”, *Proceedings of ISLPED*, 1999.

[14] B. Noble, M. Satyanarayanan, G. Nguyen, R. Katz, ”Trace-based Mobile Network Emulation *Proceedings of ACM SIGCOMM*, 1997.

[15] A. Karlin, M. Manesse, L. McGeoch and S. Owicki, “Competitive Randomized Algorithms for Nonuniform Problems”, *Algorithmica*, pp. 542–571, 1994.

[16] H. Taylor and S. Karlin, *An Introduction to Stochastic Modeling*, Academic Press, 1998.

[17] S. Ross, *Stochastic Processes*, Wiley Press, 1996.

[18] M. Berkelaar, *www.cs.sunysb.edu /algorith /implement /lpsolve /implement.shtml*

[19] J. M. Rulnick and N. Bambos, “Mobile power management for maximum battery life in wireless communication networks,” Proc. IEEE INFOCOM '96, 1996.

[20] M. Zorzi, R. Rao, “Energy-Constrained Error Control for Wireless Channels,” IEEE Personal Communications, Dec. 1997.

[21] T. S. Rappaport, “Wireless Communications,” Prentice Hall PTR, 1996.

[22] L. C. Yun and D. G. Messerschmitt, “Power control for variable QoS on a CDMA channel,” Proc. IEEE MILCOM, Forth Monmouth, NJ, 1994, pp. 178-182.