Packet Transmission Policies for Battery Operated Communication Systems

Ye Li & Chaitali Chakrabarti

Department of Electrical Engineering
Arizona State University
Tempe, AZ 85287-5706

ABSTRACT

In this paper, we address the problem of designing battery-friendly packet transmission policies for wireless data transmission. Our objective is to maximize the lifetime of battery for wireless devices subject to certain delay constraints. We present three packet transmission schemes and evaluate them with respect to battery performance. The first scheme based on combining multiple packets, utilizes the battery charge recovery due to long idle periods. The second scheme based on a modified version of lazy packet scheduling, draws lower current and is battery efficient. The third scheme which is based on a combination of the above two schemes, has superior battery performance at the expense of larger average packet delay. All three schemes were simulated for a wireless communication framework with internet traffic, and the results validated.

1. INTRODUCTION

Battery-operated portable devices are currently widely used in mobile computing for both voice and data transmission. However, portable devices cannot be used indiscriminately because they are powered by batteries which are limited energy sources. In this paper we focus on packet transmission schemes that maximize the battery lifetime.

Maximizing battery lifetime is a particularly difficult problem due to the non-linearity of the battery behavior. In recent years, there has been significant amount of work done in studying battery characteristics [4], [6] and using those characteristics to develop battery-aware task scheduling algorithms [10], [7]. All of these algorithms shape the load current profile in accordance with the battery non-linearity. A scheduling scheme that adjusts the delay of different system components of a communication system such that the discharge profile is battery-friendly has been proposed in [11].

There has also been some effort in designing battery-aware packet transmission schemes [1], [2], [3]. The basic idea is to queue transmission requests whenever the battery's state of charge (SOC) drops to a certain threshold. The schemes in [1] -- [3] use a stochastic battery model and show how queuing the requests lets the battery recover and results in overall battery lifetime enhancement.

Another important effort in energy-efficient transmission over wireless link, referred to as lazy packet scheduling [12], turns out to be battery-aware. The basic idea is to make full use of inter-arrival time between packets and transmit over a longer transmission time at reduced power. Transmitting at reduced power translates to transmitting at lower current, which results in better battery performance.

In this paper, we present three packet transmission schemes and evaluate them with respect to battery performance. The first scheme is based on combining multiple packets – larger the number of packets that are combined, better is the battery performance. This comes at the price of larger buffer size and larger system delay. The second scheme is based on a modified version of lazy packet scheduling. It has superior battery performance and lower runtime complexity. The third scheme is a combination of the first two schemes. Packets are grouped according to the first scheme, and then, transmitted following the modified lazy packet scheme. This scheme results in superior energy and charge savings. The drawback of these schemes is the increase in the packet delay. Delay analysis shows that as *RAR* reduces, the average delay per packet of the first scheme reduces mildly while that of the second and third scheme increases. In fact, the third scheme has a delay that is larger than the sum of the delay of the first and second schemes.

The remainder of the paper is organized as follows. Section 2 introduces the background of our work, which includes system configuration, the analytical battery model, and battery characteristics. Section 3 describes the packet combining scheme along with the simulation results. Section 4 discusses battery performance of the lazy packet scheme and modified lazy packet scheme. Section 5 combines the two schemes and proposes the most battery-friendly scheme. Section 6 compares and analyzes the delays resulting from the three schemes. This paper ends with a conclusion in Section 7.

2. BACKGROUND

System Configuration: As is shown in Fig.1, the communication node consists of a data processing unit, an input buffer, an output buffer and a smart battery. The data processor is used for command execution and packet processing. The smart battery supplies power to the data processing unit and buffer. It also monitors the battery's State of Charge (SOC) (defined in Section 2.3). The input and output buffers are used for temporarily storing packets before and after their processing.

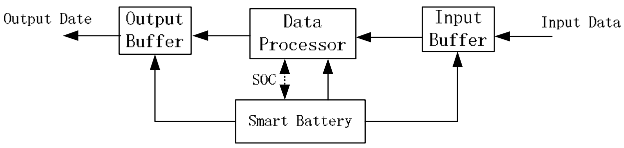

Fig.1 Communication node configuration

Battery model: In our work, we use an accurate analytical charge-based model for Li-ion batteries [8]. In this model, the load profile is given in the form of a sequence of N constant current values $I_1, I_2, I_3, ..., I_N$, where I_k is the current of task k at time t_k, and is applied for a duration $\Delta_k = t_{k+1} - t_k$. The relation between the load profile $\{ I_k, t_k \}$ and the battery's lifetime L is as follows:

$$\alpha = \sum_{k=1}^{N} I_k \Delta_k + \sum_{k=1}^{N} 2 I_k \sum_{m=1}^{\infty} \frac{e^{-\beta^2 m^2 (L - t_k - \Delta_k)} - e^{-\beta^2 m^2 (L - t_k)}}{\beta^2 m^2} \tag{1}$$

where α and β are battery parameters. The parameter α represents the total charge in the battery when it is fully charged. The parameter β measures the nonlinearity of the battery and tells us how fast the diffusion process can keep up with the rate of withdrawal of the charge. The higher the value of β, the better the battery behaves. This model has been extensively tested on real load profiles running on Compaq ITSY pocket computer. Differences between the model predictions and the measured and simulated values were in the range of 1%--3% [8]. In our simulation, we have used $\alpha = 35220$ mA-min and $\beta = 0.637$ min^{-2} corresponding to the Li-ion battery used in the ITSY pocket computer.

Since in this paper, we focus on battery-efficient packet transmission policies, we define a new parameter—battery charge consumption σ, which is the value of charge depleted minus charge recovered. In order to find the battery charge consumptions σ after execution of *M* tasks (*M*<*N*) which last till time *T* (*T*<*L*), we substitute *N* with *M* and *L* with *T* in Equation (1).

Battery properties: An important property of battery is *rate capacity*. If the current is reduced from *I* to *I/2*, the battery lasts for a time longer than *2L*, where L is the battery lifetime for current *I*. Thus lower the current load, better is the battery performance. This is the property that results in the lazy packet scheduling being battery-efficient.

Another important property of the battery is the ability to recover part of the lost charge, referred to as the *recovery effect*. As the load draws current from the battery, the electroactive species diffuse toward an electrode and a non-zero concentration gradient of the electroactive species develops across the electrolyte. If the load is now turned off, the diffusion process tends to balance the electrolyte concentration and the concentration gradient eventually equals to zero again, but at a lower level of charge. This is the property that is used to increase the battery lifetime at the expense of increase in delay in [1]—[3].

3. SCHEME 1: PACKET COMBINING

3.1 Main Features

In this scheme, several short packets queued in the buffer are combined into a long packet before processing. The packets are combined independent of the state of the battery unlike [1], [2] where packets are queued in the buffer only when the battery state of charge falls below a threshold. Figure 2(a) describes a typical transmission scenario where the packet transmission time is interspersed with idle time. Figure 2(b) shows the configuration after combining the idle time. The battery performs better if the idle time is combined together resulting in a long stretch of task time followed by a long stretch of idle time. This can be proved analytically but has not been included for lack of space.

Combining the short packets also results in a reduction in the start-up energy, since the start-up energy is used only once and not multiple times [5].

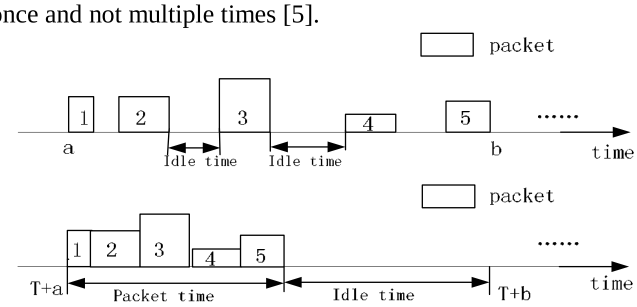

Fig. 2 (a) Original packet configuration (b) Packet configuration after combining the idle time (T: buffer delay)

3.2 Simulation Results

In this work, we assume a wireless communication system which transmits and receives internet data at a maximum speed of about 100 *Kbyte per second*. The maximum arrival rate $\lambda = 250$ *packets per second*. Here we define a new parameter—Request Arrival Rate (*RAR*), which is the ratio of instant packet arrival rate and the maximum packet arrival rate. Packets arrive according to

Poisson distribution with a loading factor of *RAR* = 0.8 (i.e. an arrival rate of 200 packets per second). Since the average packet time is 4 ms, the service rate μ = 250 *packet/second*. We also assume that the buffer size is large enough. The simulation environment is as follows.

Packet length = 400 bytes
Average packet transmission time = 4 ms
Total number of packets = 10,000$\times$ *RAR*
Average packet transmission current =200 mA
Start-up current = 780 mA
Start-up time = 0.45 ms [5]

We simulate two cases: in the first case, the internet packet time and idle time are interspersed. As a result, there is a start-up current at the beginning of every packet transmission. In the second case, internet packets and the corresponding idle time are grouped together.

Since average packet time is equal to 4 ms, the maximum number of packets in the input buffer is decided by the buffer delay. However, the average number of combined packets is equal to the maximum number of packets in input buffer multiplied by *RAR*. For example, if buffer delay is 40 ms, the maximum number of packets is 10; if *RAR* is 80%, the average number of packets in input buffer is about 10 packets$\times$ 80% = 8 packets.

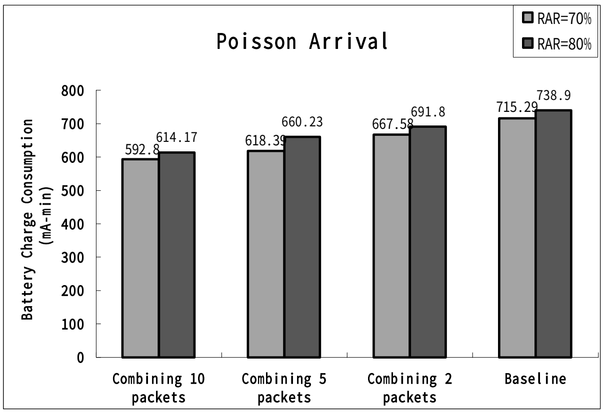

Fig. 3 Packet combining scheme: charge comparison

Fig. 3 shows the effect of packet combination on the battery charge consumption. We point out certain trends. First, the more the average number of packets combined, the lower is the battery charge consumption and larger is the delay. For *RAR* is equal to 70%, the battery charge consumption drops by 17.1% when 10 packets are combined, while the battery charge consumption drops by only 6.7% when 2 packets are combined, compared with the original configuration. This phenomenon is due to the decreasing effect of start-up energy and the larger recovery due to combined idle time. Second, as the *RAR* reduces, the average number of packets that are being processed reduces and thus the battery charge consumption reduces. Finally, the number of packets that are to be combined is determined off-line and is a compromise between the delay constraints and the charge consumption.

4. SCHEME 2: LAZY PACKET SCHEME

Lazy packet scheduling is an energy-efficient transmission scheme over a wireless link that was proposed in [12]. Lazy packet scheduling is motivated by the fact that in many channel coding schemes, the energy required to transmit a packet can be significantly reduced by lowering the transmission power and transmitting the packet over a longer period of time. The schedule is called lazy because it fully exploits all the idle slots between packets arrival times and slows down transmission. When a packet is transmitted with lower power, the current drawn by the battery is lower, resulting in better battery performance as well.

4.1 The model

Assume there are *M* packets which are processed in the time interval [*0*, *T*]. The packets have equal length. The packet inter-arrival times are denoted by d_i and the packet arrival time are denoted by t_i (see Fig. 4). Define $d_M = T - t_M$, and $\sum_{i=1}^{M} d_i = T$. Let $\vec{\tau} = [\tau_1,..., \tau_M]$ be the transmission duration of the *M* packets obtained by a schedule. All packets must meet the deadline constraint *T*.

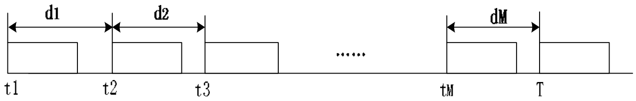

Fig. 4 Lazy packet scheme: packet arrivals in [*0*, *T*]

4.2 Online algorithm of [12]

Since lazy packet scheme extends the packet transmission time, there are some packets that have to be stored in a buffer. The online algorithm considers the number of packets stored in the buffer and calculates the transmission time based on the amount of backlog.

Assume the j th packet starts transmission at time T_j ; m_i is the transmission duration of the i th ($1 \le i \le j-1$) packet. Then

$$T_j = \sum_{i=1}^{j-1} m_i \tag{2}$$

The number of packets in buffer b_j , is given by

$$b_j = \max\{k : \sum_{i=1}^{k-1} d_i < T_j\} - j \tag{3}$$

The transmission time of a packet that starts being transmitted at time $t < T$ when there are *b* packets in buffer, can be calculated by the following formula:

$$\tau(b,t) = \max_{k\in\{1,\ldots,M\}} \{\frac{1}{k+b}\sum_{i=1}^{k} d_i\} \qquad (4)$$

where d_i are the inter-arrival times of M packets that will arrive in (t,T).

4.3 Modified lazy packet scheme

Lazy packet scheduling algorithm assigns longer packet transmission times first. This means that lower current tasks are scheduled before heavier ones most of the time. Such a profile is not battery-friendly because of the increasing current profile. Furthermore, lazy packet scheduling involves implementation of Equations (2)--(4) at run time which also causes significant charge drain. The modified lazy packet scheme equalizes the packet transmission time to avoid increasing current profile.

Let M packets that will arrive in $(0, T)$. The new transmission time is

$$\tau = \frac{T}{M} \qquad (5)$$

If the packet current of the original scheme is constant and the transmission time of the modified lazy scheme is constant (Equation (5)), the current profile due to the modified lazy scheme is flat. The flat current profile results in better battery performance compared to the increasing current profile due to lazy scheme of [12]. However, the actual difference in the battery performance is very small as will be apparent in the next section. The main advantage is in the cost of implementation is far lower (Compare equation (4) and (5)). This makes the modified scheme particularly attractive since run time costs of on-line algorithms are an important design metric.

4.4 Simulation results

In this section, we first compare the energy expended by the modified lazy packet schedule, the lazy packet schedule and the baseline schedule. Then we compare the charge consumed by the three schemes.

We choose a fixed source packet length L (400 Bytes) and a fixed packet current (200 mA) in the following simulations. Packets arrive according to Poisson distribution. The number of transmissions per bit s is calculated as follows. If we assume the transmission rate to be equal to 10^6 transmissions per second [12], and the packet transmission time to be t, we can calculate s by

$$s = \frac{t}{L}\times 10^6 = \frac{t}{400\times 8}\times 10^6 = 312.5\cdot t \qquad (6)$$

Thus, from [12] we can calculate the energy consumed for one packet by the following formula:

$$w(s) = LsN(2^{\frac{2}{s}}-1) = 10^6\times t\times(2^{\frac{2}{312.5t}}-1) \qquad (7)$$

In order to use the battery model, the change in the transmission energy has to be mapped onto the change in the packet current. If we assume the voltage to be constant, the original transmission time and packet current to be s_1 and I_1, the new transmission time and packet current using lazy packet scheme to be s_2 and I_2, then the energy of the two schemes is related by

$$\frac{W_1}{W_2} = \frac{I_1 V s_1}{I_2 V s_2} = \frac{10^6 s_1 (2^{\frac{2}{312.5 s_1}}-1)}{10^6 s_2 (2^{\frac{2}{312.5 s_2}}-1)} \qquad (8)$$

The current I_2 is related to I_1 by

$$I_2 = I_1 \times (2^{2/312.5 s_2}-1) \Big/ (2^{2/312.5 s_1}-1) \qquad (9)$$

To compare the energy and battery charge consumption of the baseline schedule, lazy packet schedule and modified lazy schedule, we choose five typical packet request arrival rate, range from 30% to 90%. Assume the total packet number is 10,000, and every time we schedule 100 packets (i.e. M = 100). Fig. 5 plots the normalized energy for different arrival rates. We see that lazy packet schemes consistently have higher energy savings. When the *RAR* is lower, the two lazy schemes save more energy. This is because at lower *RAR*, the idle times are longer and the lazy packet scheme utilizes them to extend the transmission time and thereby lower energy. The difference between the saving obtained by the modified lazy scheme and the original lazy scheme in [12] is very small. However, as pointed out earlier, the algorithm complexity of the modified lazy scheme is lower and thus the overall energy consumption, including computation energy, is lower.

To compare the computation energy consumption, we simulated the two schemes on Wattch [13], an architecture simulator that estimate the computation energy of CPU. Simulation results show that the original lazy packet scheme consumes about 329 times computation energy as that of our modified version. Note that the computation and transmission energy have been modeled by two different systems and as a result not added together to estimate the total energy.

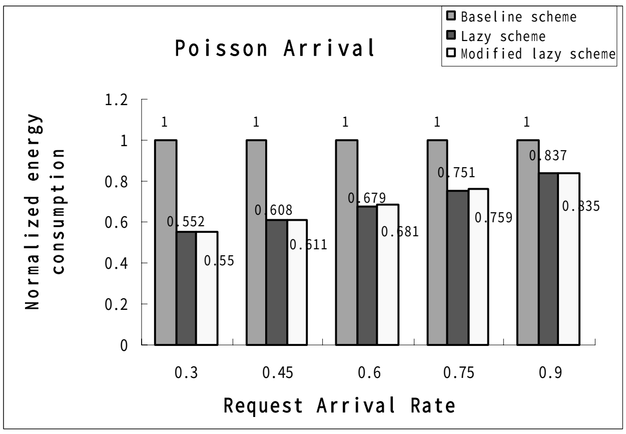

Fig. 5 Lazy packet scheme: energy consumption comparison

Next, we compare the battery charge consumption of the three schedules. Fig. 6 shows that the trend in battery charge consumption is similar to that of energy consumption. Since the two lazy packet schemes modify the original profile with short, heavy tasks with idle time to a profile with long continuous tasks with lower load, they are both battery efficient. From Fig.5 and Fig.6, we find that lazy scheme and modified lazy scheme have comparable energy and charge performance if we ignore computation energy. The modified lazy scheme has much lower overall current consumption due to significantly less on-line computation and thus better battery performance.

From the above simulations, we conclude that when the data transmission rate requirement is low or when certain packet delay is allowed, the two lazy packet schemes are both energy-efficient and battery-friendly. However, the modified one is better in terms of both energy and battery charge consumption when the computation cost of implementing the algorithm is also considered.

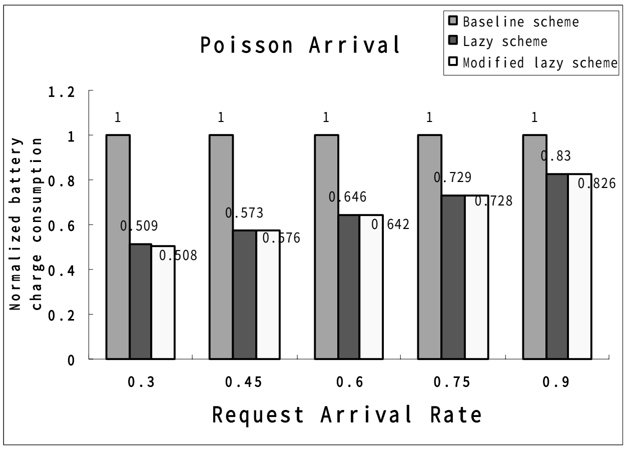

Fig. 6 Lazy schemes: Battery charge consumption comparison

5. SCHEME 3: COMBINING PACKETS + MODIFIED LAZY SCHEDULING

5.1 Main features

Recall that the combining packet transmission scheme combines several interspersed packets into a longer packet followed by a longer idle time. This idle time can be exploited by the modified lazy packet scheme which can now transmit at a lower power. The combined scheme essentially does that -- it first combines the packets and then transmits them using modified lazy scheduling.

5.2 Simulation results

We simulate this scheme using the same simulation environment as before. We do simulations under three typical packet Request Arrival Rates. We choose to combine 5 packets at a time and schedule 20 long packets (or 100 short packets) in every time slot. Fig. 6 plots the battery charge consumption for the different packet transmission schemes. All values are normalized with respect to the original scheme. From Fig. 6, we see that modified lazy scheme performs better than the packet combining scheme all the time; the difference is larger with lower *RAR*s. For instance, when *RAR* is high (for example 90%), there is only 12.3% charge difference; while when the *RAR* is low (such as 60%), there is 18.5% charge difference.

Fig. 6 also shows that the charge difference between the original scheme and the combined scheme increases as *RAR* reduces. When *RAR* is high and there is not much idle time, the packet transmission rate cannot be reduced much and the charge reduction due to the different packet combining schemes is limited. For example, when *RAR* is 90%, the combined scheme consumes 81.5% of the original charge. But when *RAR* is 45%, the combined scheme consumes only 51% of the original battery charge consumption.

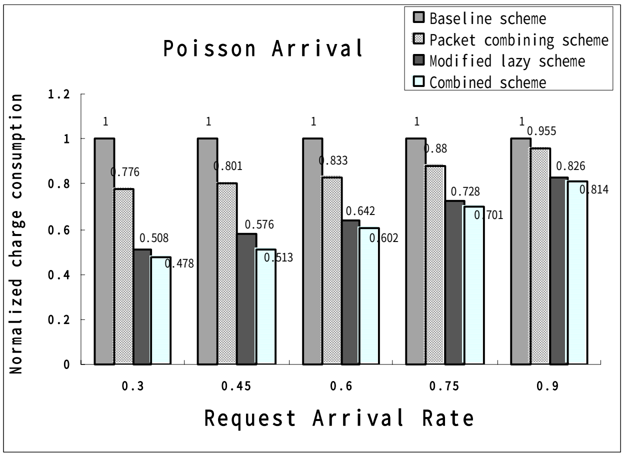

Fig. 7 Battery charge consumption comparison of the different packet transmission schemes

6. DELAY ANALYSIS

In this section, we analyze the delays for the three packet transmission schemes. Here we define the delay as the time difference between the packet arrival and the time when the packet transmitted. We assume that for the baseline scheme, the average packet delay is zero. For example, Fig.8 describes the delay of packet 1 for the three schemes. We do not take into account retransmission delays.

Fig.9 describes the average delay for the packet combining scheme for three different values of *RAR* for the case when 5 packets are combined in Poisson traffic model.

We see that the delay of the packet combining scheme reduces when the *RAR* reduces. Higher the *RAR*, longer is the delay. Since *RAR* < 1, the average delay is less than the worst case delay of 5×4 = 20 ms = T (5 packets with 4ms transmission time per packet). For instance, the average delay in Fig.8 is lower than the worst case delay (which is that of packet 1) since the delay of packet 2 and 3 are lower.

Fig.9 also shows that the average delay for modified lazy scheme increases with the reduction of *RAR*. The lower the *RAR*, the more the idle time and the longer the lazy transmission time. So the delay increases.

The combined scheme has an average delay that is larger than the sum of the delay of the packet combining scheme and the lazy packet scheme. This is because the delay of the lazy packet scheme increases when it is preceded by the packet combining scheme. Finally, as *RAR* reduces, the delay of the combined scheme increases. Thus for low *RAR*, while the combined scheme results in significant reduction in energy, it comes at a price of large increase in delay. Thus for delay sensitive systems, such a scheme will have to be used with caution.

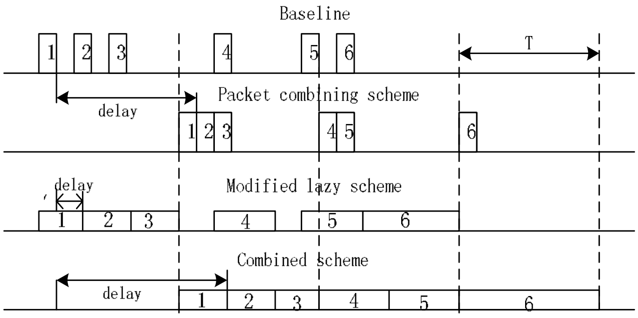

Fig.8 Definition of delay for the three schemes

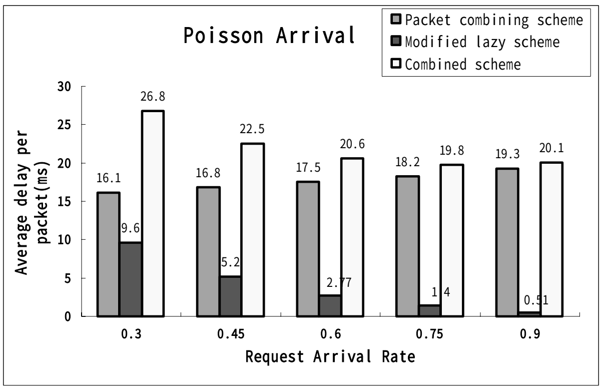

Fig.9 Delay comparison of the different packet transmission schemes

7. CONCLUSION

In this paper, we present three packet transmission schemes to increase the lifetime of battery for wireless communication devices. The first scheme is based on combining several short discontinuous packets to a longer one. The second scheme is modified lazy packet scheduling that is based on extending the transmission time and reducing the packet current. The third scheme is a combination of the two schemes. We simulate these schemes in a normal wireless link framework and demonstrate their battery efficiency. We also analyze the average delay per packet for the three schemes. We find that the modified lazy scheme has the best performance with respect to both battery charge consumption and delay.

Acknowledgements: This research was funded by Connection One, a NSF/IUCRC center.

8. REFERENCES

[1] C.F. Chiasserini, and R.R.Rao, "Improving battery performance by using traffic shaping techniques", IEEE J. Select. Areas Commun. vol.19, No.7, July 2001, pp 1385-1394.

[2] C.F. Chiasserini, and R.R.Rao, "Energy efficient battery management", IEEE J. Select Areas Commun., vol.2, July 2001, pp 396-403.

[3] C.F. Chiasserini, and R.R.Rao, "A traffic control scheme to optimize the battery pulsed discharge", IEEE Select. Areas MILCOM., vol.2, 1999, pp 1419-1423.

[4] T. Martin, "Balancing batteries, power, and performance: System issues in CPU speedsetting for mobile computing", Ph.D Dissertation, 1999.

[5] S. Cho and Anantha P. Chandrakasan, "Energy efficient protocols for low duty cycle wireless microsensor networks", ICASSP'01, vol.4, 2001, pp 2041-2044.

[6] D. Rakhmatov, "Modeling and optimization of energy supply and demand for portable reconfigurable electronic systems", Ph.D Dissertation, 2002.

[7] J. Luo, and N.K. Jha, "Battery-aware static scheduling for distributed real-time embedded system." DAC 2001, pp 444-449.

[8] D. N.Rakhmatov and S. B.K. Vrudhula, "An analytical high-level battery model for use in energy management of portable electronic system", ICCAD 2001, pp 488-493.

[9] M. Zorzi and R. R.Rao, "ARQ error control for delay-constrained communications on short-range bursty-error channels", Vehicular Technology Conference, 1997 IEEE 47th, vol.3, 1997, pp 1528-1532.

[10] P. Chowdhury and C. Chakrabarti, "Battery aware task scheduling for a system-on-chip using voltage/clock scaling", SIPS 2002, IEEE, pp 201-206

[11] K. Lahiri, A. Raghunathan and S. Dey, "Communication based power management", Design&Test of Computers, IEEE, Vol.19, 2002, pp 118-130.

[12] B. Prabhakar, E. U. Biyikoglu and A. E. Gamal, "Energy-efficient transmission over a wireless link via lazy packet scheduling", INFOCOM 2001, pp 386-394.

[13] D. Brooks, V. Tiwari and M. Martonosi, "Wattch: a framework for architectural-level power analysis and optimizations", Computer Architecture 2000, pp 83-94.