Energy-efficient Transmission over a Wireless Link via Lazy Packet Scheduling

Balaji Prabhakar, Elif Uysal Biyikoglu and Abbas El Gamal
Information Systems Laboratory
Stanford University
Stanford, CA 94305
balaji@isl.stanford.edu, elif@stanford.edu, abbas@isl.stanford.edu

Abstract—**The paper considers the problem of minimizing the energy used to transmit packets over a wireless link via *lazy* schedules that judiciously vary packet transmission times. The problem is motivated by the following key observation: In many channel coding schemes, the energy required to transmit a packet can be significantly reduced by lowering transmission power and transmitting the packet over a longer period of time. However, information is often time-critical or delay-sensitive and transmission times cannot be made arbitrarily long. We therefore consider packet transmission schedules that minimize energy subject to a deadline or a delay constraint. Specifically, we obtain an optimal offline schedule for a node operating under a deadline constraint. An inspection of the form of this schedule naturally leads us to an online schedule which is shown, through simulations, to be energy-efficient. Finally, we relax the deadline constraint and provide an exact probabilistic analysis of our offline scheduling algorithm. We then devise a lazy online algorithm that varies transmission times according to backlog and show that it is more energy efficient than a deterministic schedule that guarantees stability for the same range of arrival rates.**

I. Introduction

Ubiquitous wireless access to information is gradually becoming reality. Dedicated-channel voice transmission (as in most existing cellular systems, e.g. GSM, IS-95) has already become a widespread and mature technology. Packet-switched networks are being considered for heterogeneous data (combined voice, web, video, etc.) to efficiently use the resources of the wireless channel. Wireless LANs and personal area networks, where packetization is more suited to the bursty nature of the data, are being developed and deployed. More recently, ad-hoc networks and networks of distributed sensors are being designed utilizing the robustness and asynchronous nature of transmissions in packet networks.

A key concern in all of these wireless technologies is energy-efficiency. The mobility of a hand-held wireless device is limited by the fact that its battery has to be regularly recharged from a power source. In a sensor network, the sensors may not be charged once their energy is drained, hence the lifetime of the network depends critically on energy. It has therefore been of wide interest to develop low power signaling and multiaccess schemes, signal processing circuits and architectures to increase battery life.

There has been a lot of research on transmission power control schemes over the past few years (see, for example,[3], [8], [10], [11], [13], [16], [18]). The chief motivation of these schemes, however, has not been to directly conserve energy but rather to mitigate the effect of interference that one user can cause to others. The specific aims ranged from obtaining distributed power control algorithms to determining the information theoretic capacity achievable under interference limitations ([1],[12]).

Whereas most power control schemes aim at maximizing the amount of information sent for a given average power constraint, a recent study that we are aware of ([2]) considers minimizing the power subject to a specified amount of information being successfully transmitted. Rather than minimizing power, [5] considers the question of minimizing energy directly; and compares the energy efficiency, defined as the ratio of total amount data delivered and total energy consumed, of several medium access protocols.

In this paper we consider the problem of minimizing the energy used by a node in a wireless data network to transmit packetized information within a given amount of time. The setup attempts to model a number of realistic wireless networking scenarios. (i) A node with finite lifetime and finite energy supply such as in a sensor network [14]. (ii) A battery operated node with finite-lifetime information; that is, information that must be transmitted before a deadline. (iii) A battery operated node that is periodically recharged. In this case minimizing transmission energy ensures that the node does not run out of energy before it is recharged.

To minimize transmission energy we vary packet transmission times and power levels to find the optimal schedule for transmitting the packets within the given amount of time. The idea of minimizing energy by varying packet transmission time is based on the following key observation: It is possible to reduce energy by lowering transmission power and transmitting the packet over a longer period of time. Clearly more time cannot hurt, since one can always do nothing during the extra time. But one can do much better– examples in the the next section demonstrate that many channel coding schemes can take advantage of this extra time to significantly reduce transmission energy. In particular, the examples suggest that when the channel is time-invariant the packet transmission energy is strictly decreasing and convex in transmission time.

The above discussion implies that it makes sense to transmit information over a longer period of time to conserve energy. However, since all packets must be transmitted within the given amount of time, the transmission time of any one packet cannot be arbitrarily long as this would leave too little a time for the transmission of future packets and increase the overall energy spent. The rest of the paper attempts to understand this trade-off precisely, and to exploit it to devise energy-efficient schedules.

Outline of paper

In Section II we set up the framework for modeling the minimum energy packet transmission scheduling problem for a node with a finite lifetime T and introduce notations to be used throughout the paper. In Section III we find the energy-optimal offline transmission schedule. We call this schedule *lazy* since it fully exploits any lulls in packet arrival times to conserve energy by slowing down transmission. In Section IV we formulate the optimal offline schedule in a manner that naturally suggests an online schedule. We show that this online schedule is quite energy efficient – it achieves average energy that is surprisingly close to the optimal offline algorithm. The comparison is done using simulations since it is hard to conduct analytical comparisons for finite T.

By letting $T \to \infty$ and assuming Poisson arrivals, we are able to conduct an exact analysis of the optimal offline schedule (as outlined in the Appendix). This not only gives us insight into how to design energy-efficient online schedules, but also suggests a formulation of the online scheduling problem where we seek to minimize transmission energy subject to average packet delay constraint, instead of the deadline constraint T. Under this formulation we find that a lazy schedule that transmits packets for longer times when the backlog is low and for shorter times when the backlog is high outperforms the deterministic schedule which uses a fixed packet transmission time for all packets.

This is an interesting comparison because among schedules that are independent of the packet arrival process (and hence are oblivious of backlogs), the deterministic schedule achieves the smallest average delay[1], which implies that it has the highest transmission times, and hence the lowest energy. The fact that lazy schedules are more energy-efficient than the deterministic schedule, therefore, demonstrates the need to take advantage of lulls in packet arrival times.

[1]By the well-known folk theorem "determinism minimizes delay" [17].

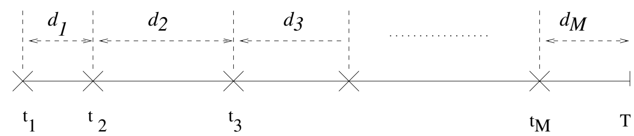

Fig. 1. Packet arrivals in $[0, T)$

Finally, Section V outlines further work and concludes the paper.

II. THE MODEL AND NOTATION

We begin by modeling a node whose lifetime is finite, equal to T, say. Consider a transmitter-receiver pair. The transmitter needs to send a random number, M, of packets that arrive in the time interval $[0, T)$ (see Figure 1). The packets are all assumed to be of equal length[2]. In the figure, the arrival times of packets, t_i, are marked by crosses and inter-arrival epochs are denoted by d_i. Without loss of generality, we may assume that the first packet is received at time 0. For convenience we define $d_M = T - t_M$, and hence $\sum_{i=1}^{M} d_i = T$. Let $\vec{\tau} = (\tau_1, \tau_2, \ldots, \tau_M)$ be the transmission durations of the packets as obtained by a schedule. All packets must be transmitted to the receiver within $[0, T)$.

Definition 1: A vector $\vec{\tau} = [\tau_1 \ldots \tau_M]$ of transmission durations is *feasible* if
(i) For $1 \leq k < M$, $\sum_{i=1}^{k} \tau_i \geq \sum_{i=1}^{k} d_i$.
(ii) $\sum_{i=1}^{M} \tau_i \leq T$.
In words, no packet may begin transmitting before it has arrived and all transmissions must be over by time T.

We seek an answer to the question: How should the scheduler choose $\vec{\tau}$ so that the total *energy* used to transmit all the packets is minimized?

Let $w(\tau)$ denote the energy required to transmit one packet over a duration τ. The only assumptions we make about $w(\tau)$ are:
(i) $w(\tau) \geq 0$.
(ii) $w(\tau)$ is monotonically decreasing in τ.
(iii) $w(\tau)$ is strictly convex in τ.

Assumption (i) is obvious. We shall now demonstrate that assumptions (ii) and (iii) hold by considering the energy required to reliably transmit one bit of information over a wireless link. The following two examples assume a discrete time Additive White Gaussian Noise (AWGN) channel model for the wireless link and consider two different channel coding schemes.

1. *Optimal channel coding:* Consider an AWGN wireless channel with average signal power constraint P and noise power N. As is well known [6], the information theoretically optimal channel coding scheme, which employs

[2]This assumption is made for simplicity of presentation. The results easily extend to the general case of variable packet lengths.

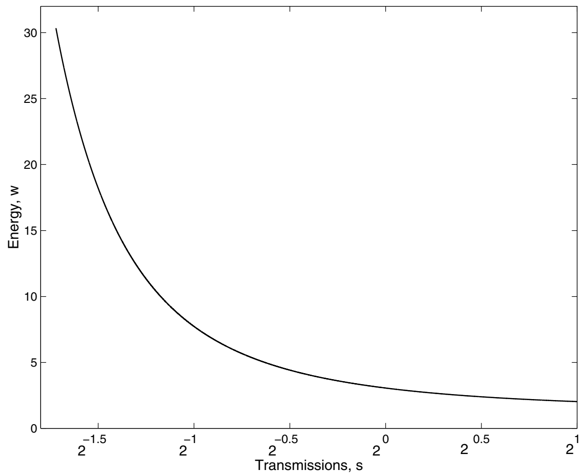

Fig. 2. Energy per bit vs. transmission time with optimal coding.

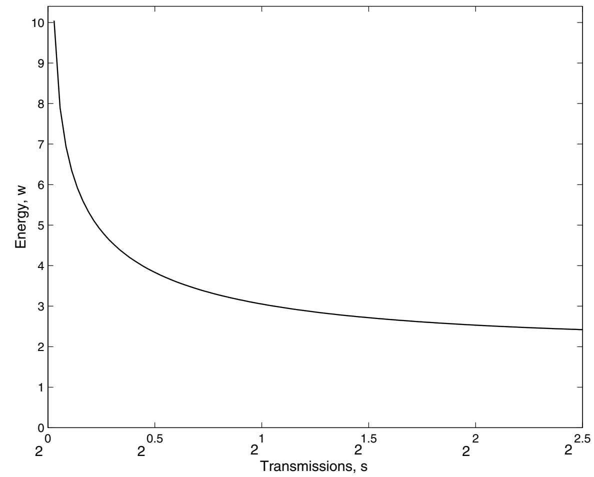

Fig. 3. Energy per bit vs. transmission time for the suboptimal coding scheme.

randomly generated codes, achieves the Shannon channel capacity given by

$$C_1 = \frac{1}{2}\log_2\left(1+\frac{P}{N}\right) \text{ bits/transmission.} \quad (1)$$

More precisely given any $0 < \alpha < 1$ information can be reliably transmitted at rate $R = \alpha C_1$. To determine the energy per bit w, note that $s = \frac{1}{R}$ can be interpreted as the number of transmissions per bit and substitute in equation (1) to get

$$w = sP = sN(2^{\frac{2}{\alpha s}} - 1).$$

It is easy to see that w is *monotonically decreasing and convex* in s, and that as s approaches infinity the energy required to transmit a bit, $w_\infty = \frac{2}{\alpha}\ln 2 = 1.3863$. Figure 2 plots w vs. s for $N = 1$ and $\alpha = 0.99$. The range of s in the plot corresponds to SNR values from 20dB down to 0.11dB. This is a typical range of SNR values for a wireless link. In this range w can be decreased by a factor of 20 by increasing transmission time and correspondingly decreasing power.

2. *A suboptimal channel coding scheme:* To show that our observation holds for other, suboptimal channel coding schemes, we consider scheme using antipodal signaling [15] and binary block error correction coding again over an AWGN wireless link. It can be shown that the minimum error probability per bit using antipodal signaling over an AWGN channel is given by

$$p = Q\left(\sqrt{\frac{P}{N}}\right),$$

where Q is the well known Gaussian Q-function. Using this signaling scheme the channel is converted into a binary symmetric channel (BSC) with cross over probability p. The optimal binary error correction coding scheme achieves the Shannon capacity for the BSC, given by

$$C_2 = 1 - h(p) \text{ bits/transmission,}$$

where $h(p)$ is the binary entropy function. Thus for any $0 < \alpha < 1$, information can be reliably transmitted at rate $R = \alpha C_2$. Again interpreting $s = \frac{1}{R}$ to be the number of transmitions per bit, the energy per bit can be computed as a function of s as depicted in Figure 3 for $N = 1$ and $\alpha = 0.99$. Note that again w is *monotonically decreasing and convex* in s, and converges to a limit $w_\infty = 2.108$, which is, as expected, larger than that using optimal coding. The range of s in the figure corresponds to SNR between 20dB to -3.7dB. In this range w drops by over a factor of 8.

The assumption that energy is a decreasing function of transmission duration implies that the schedule must allow each packet to be transmitted over as long a time as possible. However, since all packets must be transmitted within T units of time, the transmission time of any one packet cannot be arbitrarily long as this will leave too little a time for the transmission of future packets and increase the overall energy spent. But, it is possible that no packets may arrive in the future and therefore the node may well have transmitted the current packet until time T!

It is precisely this tension we wish to address. We first consider the problem of designing an optimal "offline" algorithm, assuming that all arrival times are known at time 0. The structure of the optimal offline schedule will give us guidelines for designing online algorithms which, by definition, make their decisions without knowledge of the future.

III. Optimal Offline Scheduling

In this section we determine the energy-optimal offline schedule. After briefly introducing the basic setup, a necessary condition for optimality is stated (Lemma 1). This motivates the definition of a specific schedule $\vec{\tau^*}$ (Definition 2). The schedule $\vec{\tau}^*$ is shown to be feasible (Lemma 2), and the energy-optimal offline schedule (Theorem 1).

Suppose that the arrival times $t_i, i = 1, \ldots, M$ of M packets, which arrive in the interval $[0, T)$, is known at time 0. As before, assume that $t_1 = 0$. The problem is to determine $\vec{\tau}$, the vector of transmission times, so as to minimize $w(\vec{\tau}) = \sum_{i=1}^{M} w(\tau_i)$.

The assumption that $w(\tau)$ decreases with τ trivially implies it is sub-optimal to have transmission times such that $\sum_i \tau_i < T$. For, we could simply increase the transmission times of one or more packets and reduce $w(\vec{\tau})$. Hence we will only consider transmission schedules $\vec{\tau}$ which satisfy $\sum_i \tau_i = T$.

Lemma 1: A necessary condition for optimality is

$$\tau_i \geq \tau_{i+1} \quad \text{for} \quad i \in \{1, \ldots, M-1\}. \tag{2}$$

Proof. Let $\vec{\tau}$ be a feasible vector such that $\tau_i < \tau_{i+1}$ for some $i \in \{1, \ldots, M-1\}$. Further suppose that it is optimal. Consider the schedule $\vec{\sigma}$ such that $\sigma_i = \sigma_{i+1} = \frac{\tau_i + \tau_{i+1}}{2}$ and $\sigma_j = \tau_j$ for $j \neq i, i+1$. It is easy to verify that $\vec{\sigma}$ is feasible. Comparing the energies used by $\vec{\tau}$ and $\vec{\sigma}$ we obtain

$$\begin{aligned} w(\vec{\tau}) - w(\vec{\sigma}) &= w(\tau_i) + w(\tau_{i+1}) \\ &\quad - w(\sigma_i) - w(\sigma_{i+1}) \\ &= w(\tau_i) + w(\tau_{i+1}) \\ &\quad - 2w(\frac{\tau_i + \tau_{i+1}}{2}) \\ &\stackrel{(a)}{>} 0, \end{aligned}$$

where inequality (a) follows from the strict convexity of $w(\cdot)$. This contradicts the optimality of $\vec{\tau}$ and proves the lemma. ■

The proof of the above lemma suggests the form of the optimal offline schedule: Equate the transmission times of each packet, subject to feasibility constraints. We proceed to do just this and define the optimal schedule next.

Given packet inter-arrival times $d_i, i \in \{1, \ldots, M\}$, let $k_0 = 0$, and define

$$\begin{aligned} m_1 &= \max_{k \in \{1, \ldots, M\}} \{\frac{1}{k} \sum_{i=1}^{k} d_i\} \text{ and} \\ k_1 &= \max\{k : \frac{1}{k} \sum_{i=1}^{k} d_i = m_1\}. \end{aligned}$$

For $j \geq 1$, let

$$\begin{aligned} m_{j+1} &= \max_{k \in \{1, \ldots, M-k_j\}} \{\frac{1}{k} \sum_{i=1}^{k} d_{k_j+i}\} \text{ and} \\ k_{j+1} &= k_j + \max\left\{k : \frac{\sum_{i=1}^{k} d_{k_j+i}}{k} = m_{j+1}\right\}, \end{aligned}$$

where k varies between 1 and $M - k_j$. We proceed as above to obtain pairs (m_j, k_j) until $k_j = M$ for the first time[3]. Let $J = \min\{j : k_j = M\}$. The pairs $(m_j, k_j), j = 1, \ldots, J$ are used to define $\vec{\tau}^*$ below, and Theorem 1 shows that $\vec{\tau}^*$ is the optimal offline schedule.

Definition 2: Let $\vec{\tau}^*$ be the schedule defined as follows:

$$\tau_i^* = m_j \text{ if } k_{j-1} < i \leq k_j. \tag{3}$$

Figure 4 shows an example of this schedule. The arrivals in the figure have been randomly generated (with exponentially distributed inter-arrival intervals of mean 1) in a time window of $T = 20$. The heights of the bars are proportional to the magnitudes of the d's and τ^*'s.

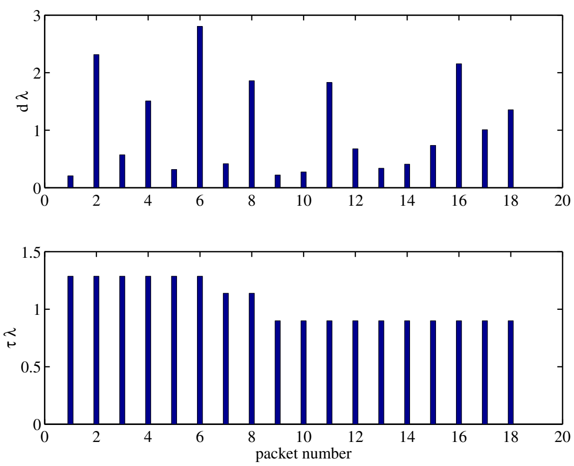

Fig. 4. An example run of d's (top) and τ^*s (bottom)

Lemma 2: The following hold for the schedule $\vec{\tau}^*$ of Definition 2:
(i) It is feasible and $\sum_{i=1}^{M} \tau_i^* = T$.
(ii) It satisfies the condition stated in Lemma 1.

Proof. We first establish (i). For $1 \leq k \leq k_1$,

$$\sum_{i=1}^{k} \tau_i^* = k\, m_1 \geq k \sum_{i=1}^{k} \frac{d_i}{k} = \sum_{i=1}^{k} d_i,$$

where the inequality follows from the definition of m_1.

Similarly for $k_1 < k \leq k_2$,

$$\begin{aligned} \sum_{i=1}^{k} \tau_i^* &= k_1\, m_1 + (k - k_1) m_2 \\ &\geq \sum_{i=1}^{k_1} d_i + (k - k_1) \sum_{i=k_1+1}^{k} \frac{d_i}{k - k_1} \\ &= \sum_{i=1}^{k} d_i. \end{aligned}$$

[3]Note that, by definition, $k_j < k_{j+1}$. Therefore the k_j are increasing with j and will equal M for some j.

Proceeding thus, we obtain that $\sum_{i=1}^{k} \tau_i^* \geq \sum_{i=1}^{k} d_i$ for all $k, 1 \leq k \leq M$.

To finish the proof of (i) it only remains to show that $\sum_{i=1}^{M} \tau_i^* = T$. Now

$$\sum_{i=1}^{M} \tau_i^* = \sum_{j=1}^{J} (k_j - k_{j-1}) m_j, \quad (4)$$

where $k_0 = 0$ and $k_J = M$. By definition of m_j and k_j, it follows that for each j

$$(k_j - k_{j-1}) m_j = \sum_{k=k_{j-1}+1}^{k_j} d_k.$$

Using this at equation (4), we get $\sum_{i=1}^{M} \tau_i^* = \sum_{i=1}^{M} d_i = T$. This establishes (i).

As for (ii), it suffices to show that $m_j > m_{j+1}$ since this implies $\tau_i^* \geq \tau_{i+1}^*$ for each i. We first show that $m_1 > m_2$. For any $k \in [k_1 + 1, k_2]$,

$$\begin{aligned} m_1 &= \frac{d_1 + \ldots + d_{k_1}}{k_1} \\ &\stackrel{(a)}{>} \frac{d_1 + \ldots + d_{k_1}}{k} + \frac{d_{k_1+1} + \ldots + d_k}{k} \\ &= \frac{k_1}{k} m_1 + \frac{(k - k_1)}{k} \frac{(d_{k_1+1} + \ldots + d_k)}{k - k_1}, \end{aligned}$$

where (a) follows from the definition of m_1. Choosing $k = k_2$, we get

$$m_1 > \frac{k_1}{k_2} m_1 + \frac{k_2 - k_1}{k_2} m_2,$$

from which it follows that $m_1 > m_2$.

In an exactly similar fashion it can be shown that $m_2 > m_3$ and, more generally, that $m_j > m_{j+1}$ for any $j, 1 \leq j \leq J - 1$. This establishes (ii) and completes the proof of the lemma. ∎

Theorem 1: The schedule $\vec{\tau}^*$ of Definition 2 is the optimum offline schedule.

Proof. Consider any other feasible schedule $\vec{\tau}$. Let i be the first index where $\tau_i \neq \tau_i^*$. We show that $w(\vec{\tau}) > w(\vec{\tau^*})$. There are two possibilities to consider.

Case 1: $\tau_i > \tau_i^*$. Since $\sum_j \tau_j = T$ (else, $\vec{\tau}$ would idle for some time, making it sub-optimal), there must be at least one $j > i$ for which $\tau_j < \tau_j^*$. Let $r = \min\{j : i < j \leq M, \tau_j < \tau_j^*\}$. Consider the schedule $\vec{\sigma}$ defined as follows:

$$\begin{aligned} \sigma_i &= \tau_i - \Delta && (5) \\ \sigma_r &= \tau_r + \Delta && (6) \\ \sigma_j &= \tau_j \text{ for all } j \neq i, r && (7) \end{aligned}$$

where $\Delta = \min\{(\tau_i - \tau_i^*), (\tau_r^* - \tau_r)\}$.

Claim 1: The schedule $\vec{\sigma}$ is feasible.
Proof of Claim 1: Since $\sum_j \sigma_j = \sum_j \tau_j = T$, the second condition for feasibility is verified. By the definition of the indices i and r, and the feasibility of $\vec{\tau}$ and $\vec{\tau^*}$, it follows that

$$\begin{aligned} \sum_{j=1}^{k} \sigma_j &= \sum_{j=1}^{k} \tau_j \geq \sum_{j=1}^{k} d_j \text{ for } 1 \leq k \leq i-1 && (8) \\ \sum_{j=1}^{i} \sigma_j &\geq \sum_{j=1}^{i} \tau_j^* \geq \sum_{j=1}^{i} d_j && (9) \\ \sum_{j=1}^{k} \sigma_j &\geq \sum_{j=1}^{k} \tau_j^* \geq \sum_{j=1}^{k} d_j \text{ for } i < k \leq r && (10) \\ \sum_{j=1}^{k} \sigma_j &= \sum_{j=1}^{k} \tau_j \geq \sum_{j=1}^{k} d_j \text{ for } k > r. && (11) \end{aligned}$$

This verifies the first condition for feasibility and proves Claim 1.

Claim 2: $w(\vec{\sigma}) < w(\vec{\tau})$.
Proof of Claim 2:

$$\begin{aligned} w(\vec{\tau}) - w(\vec{\sigma}) &= w(\tau_i) + w(\tau_r) \\ &\quad -w(\sigma_i) - w(\sigma_r) \\ &= w(\tau_i) - w(\tau_i - \Delta) \\ &\quad +w(\tau_r) - w(\tau_r + \Delta) \\ &\stackrel{(a)}{>} 0, \end{aligned}$$

where inequality (a) follows from two facts: (i) $w(\cdot)$ is strictly convex and decreasing, and (ii) $\tau_i > \tau_r$. That is, for any real-valued function f that is strictly convex and decreasing, and for any $a, b \in I\!R$ such that $a < b$, we have $f(b) - f(b-\delta) + f(a) - f(a+\delta) > 0$, where $0 < \delta < b - a$. This proves Claim 2.

Thus, under Case 1, any feasible schedule $\vec{\tau}$ may be modified to obtain a more energy efficient schedule $\vec{\sigma}$. Therefore schedules which are different from $\vec{\tau^*}$ in the sense of Case 1 are sub-optimal.

Case 2: $\tau_i < \tau_i^*$. We shall argue for a contradiction and show that such a $\vec{\tau}$ is infeasible.

From the definition of $\vec{\tau^*}$ we know that $\tau_i^* = m_j$, assuming $k_{j-1} < i \leq k_j$. In fact $\tau_l^* = m_j$ for all $k_{j-1} < l \leq k_j$.

Since i is the first index where $\vec{\tau}$ and $\vec{\tau^*}$ disagree, $\tau_l = \tau_l^*$ for all $l < i$. Suppose that the schedule $\vec{\tau}$ satisfies the condition of Lemma 1 (else it is sub-optimal and we are done). It follows that $\tau_i \geq \ldots \geq \tau_{k_j}$, and we get

$$\sum_{l=1}^{k_j} \tau_l^* > \sum_{l=1}^{k_j} \tau_l. \quad (12)$$

But, by definition of $\vec{\tau^*}$,

$$\sum_{l=1}^{k_j} \tau_l^* = \sum_{l=1}^{j} (k_l - k_{l-1})\, m_l = \sum_{l=1}^{k_j} d_l.$$

Equation (12) now gives $\sum_{l=1}^{k_j} \tau_l < \sum_{l=1}^{k_j} d_l$, implying that $\vec{\tau}$ is infeasible.

This contradiction concludes Case 2 and the proof of Theorem 1 is complete. ■

In short, lazy scheduling trades-off delay for energy. To do this, it necessarily buffers packets. The dramatic energy savings that come from simply keeping a small buffer is best illustrated by an example: Imagine a scheme that keeps a buffer size of zero (hence transmission times can at most be set equal to inter-arrival times). For the set of packet arrivals shown in Figure 4, the optimal offline schedule achieves an energy of 65.445 and the zero-buffer scheme (which, therefore, has no queuing delay) achieves an energy 77.78×10^5; five orders of magnitude larger (using an energy function $\tau(2^{\frac{6}{\tau}} - 1)$).

IV. Online scheduling

In this section we develop and evaluate energy efficient online scheduling algorithms based on the optimal offline algorithm discussed in the previous section. In order to design online algorithms that are energy efficient on *average*, one needs the statistics of the arrival process. Whilst our approach is general, for concreteness and tractability, we assume Poisson arrivals for the analysis conducted in this paper. We note that Poisson arrivals are unrealistic in the wireless LAN environment, where arrivals tend to be more bursty. In fact, we have observed that when arrivals are bursty, lazy scheduling performs even better than in the Poisson case; for, one can take advantage of a small queueing delay and greatly reduce transmission energy.

We proceed by first formulating the offline algorithm in a manner that is suited for online use (Section IV-A). Based on this formulation we propose an online algorithm (Section IV-B) and, using simulations, show that on the average it is almost as energy efficient as the optimal offline schedule (Section IV-C).

We then investigate the important special case of $T \to \infty$. In this case we are able to analyze the optimal offline schedule exactly (in the Appendix), obtain an online lazy schedule as a result of this analysis, and perform comparisons of the energy efficiency of the lazy schedule and a fixed-transmission time online algorithm (Section IV-D).

A. Online formulation of offline schedule

Consider the time interval $[0, T)$ and as before assume that a packet arrives at time 0. Suppose also that packets arrive as a Poisson process of rate λ. Conditioned on there being $M-1$ arrivals in $(0, T)$, let the inter-arrival times be denoted by D_i. Denote the optimal offline schedule for these M packets by $\vec{\tau^*}$. The time at which the j^{th} packet starts transmitting is

$$T_j^* = \sum_{i=1}^{j-1} \tau_i^*.$$

The quantity b_j, given by

$$b_j = \max\{k : \sum_{i=1}^{k-1} D_i < T_j^*\} - j,$$

is the *backlog* in the queue when the j^{th} packet starts transmitting. Observe that this backlog does not include the j^{th} packet; that is, if $b_j = 1$, then there is precisely one packet (namely, the $(j+1)^{th}$) in the queue when the j^{th} packet starts transmitting. Finally, let $C_i, i \in \{1, \ldots, M-j-b_j\}$ be the inter-arrival times between packets that arrive *after* T_j^*. Thus, when the j^{th} packet starts transmitting the situation is this: (i) The "time to go" equals $T - T_j^*$, (ii) there are b_j packets currently backlogged, (iii) $M - j - b_j$ packets are yet to arrive and the first of these will arrive in C_1 units of time, the second will arrive in $C_1 + C_2$ units of time, etc.

With this notation and some algebra, it can be shown that τ_j^* is also given by

$$\tau*_j = \max_{k \in \{1,\ldots,M-(j+b_j)\}} \{\frac{1}{k+b_j} \sum_{i=i}^{k} C_i\}. \quad (13)$$

That is, the optimal offline schedule applies exactly the same formula for computing the transmission time of each packet by taking into account the current backlog, future arrivals, and the time to go!

B. Online algorithm

The form of the optimal offline schedule, $\vec{\tau^*}$, we have obtained strongly suggests the following online algorithm: The transmission time of a packet that starts being transmitted at time $t < T$ when there is a backlog of b packets can be set equal to the *expected value* of the random variable

$$\tau(b,t) = \max_{k \in \{1,\ldots,M\}} \left\{ \frac{1}{k+b} \sum_{i=1}^{k} D_i \right\}, \quad (14)$$

where b is the current backlog, D_i are the inter-arrival times of the (random number) M of packets that will arrive in (t, T).

In the following, schedules based on $E(\tau(b,t))$ will be used. Note that these algorithms are not necessarily optimal.

To proceed, we need to evaluate $E(\tau(b,t))$. This is difficult to do analytically when T is finite. In the simulations of the next section, (Section (IV-C) $E(\tau(b,t))$ is evaluated numerically.

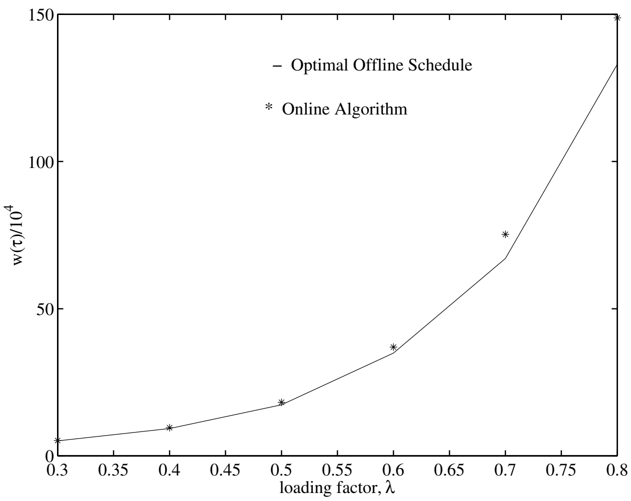

Fig. 5. A comparison of the online algorithm with the optimal offline algorithm.

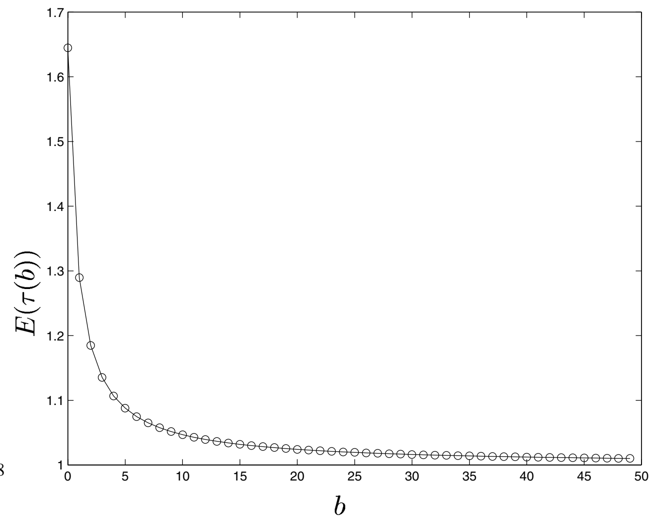

Fig. 6. A plot of $E(\tau(b))$ vs. b for $\lambda = 1$.

C. Simulations: Finite time horizon

Using simulations we compare the energies expended by the online algorithm defined above and the optimal offline algorithm. The setup is as follows. A finite time horizon $T = 10$ sec is chosen. We assume a packet length of $B = 10$ KBits and a maximum packet transmission rate of 100 packets per second. Within the time period T, we assume that packets arrive according to a Poisson process at a loading factor of $\lambda = 0.7$ (i.e. an arrival rate of 70 packets per second). Since it is possible for packets to arrive arbitrarily close to the finish time T, if we insist that these very late arrivals also be transmitted before the deadline T, then *any* algorithm, including the optimal offline algorithm, incurs a huge energy cost. This makes comparisons of performance difficult and meaningless. We therefore use a "guard band" g around the finish time and disallow packets from arriving after time $T - g$. For the comparison we use the following formula[4] for the packet transmission energy w as a function of packet transmission time τ in seconds,

$$w(\tau) = \frac{\tau 10^4}{0.06}(2^{\frac{.12}{\tau}} - 1). \quad (15)$$

Figure 5, which plots the energy per packet against transmission time, shows that the online algorithm is almost as energy-efficient as the optimal offline algorithm.

D. Infinite time horizon: Formulation and simulations.

The algorithm presented above was directly motivated by the optimal offline algorithm. It is of interest to let $T \to \infty$ and look at how the lazy schedule performs in terms of energy and delay. As an added bonus, the infinite-horizon problem turns out to be exactly analyzable when arrivals are Poisson, seemingly of independent interest. Defining $E(\tau(b)) \triangleq E(\lim_{t\to\infty} \tau(b,t))$, it is shown in the appendix that $E(\tau(b)) = \frac{(1+b)}{\lambda}(\frac{\pi^2}{6} - \sum_{k=1}^{b} \frac{1}{k^2})$. Figure 6 plots $E(\tau(b))$ as a function of the backlog b when the arrivals are a rate 1 Poisson process. As can be seen, the average transmission time of the offline schedule decreases with the backlog, approaching $\frac{1}{\lambda}$ as the backlog, b, approaches infinity.

This exact analysis of the offline algorithm not only provides us with insight into the manner in which transmission times should depend on backlog, but also suggests a specific online schedule. Unlike the finite T case where online schedules can be compared solely on the basis of their energy expenditure, when $T = \infty$ packet delays must be taken into consideration. Otherwise, energy comparisons become meaningless since we can simply let transmission times be arbitrarily long and obtain the minimum possible transmission energy per packet.

Accordingly, suppose packets arrive according to a rate λ Poisson process at a transmission node with infinite queue capacity. The node transmits a packet p for a duration $\tau(b)$ when the backlog in the queue, excluding packet p, is b. The arrival rate λ is not known at the transmitter, but it is known that $\lambda \leq \lambda_{max}$. The transmitter needs to be designed to ensure stability, and since λ_{max} is a worst case estimate of the arrival rate, stability will be ensured if the rate of transmission is higher than λ_{max}. Since a lazy schedule varies transmission times depending on the backlog according to the function $\tau(b)$, for stability it suffices that $\tau(b) < \frac{1}{\lambda_{max}}$ for all b large enough.

[4]The formula is obtained using the information theoretic capacity formula in equation (1) for the AWGN channel with noise power $N = 1$ and rate $R = 6$ (or SNR$= 36dB$) to transmit a 10Kbit packet in 10msec, i.e. reliable transmission at link speed of 1 Mbits per second.

λ	Lazy schedule		Deterministic schedule	
	Eng/pk $\times 10^{-4}$	Dly/pk$\times 10^{-1}$	Eng/pk $\times 10^{-4}$	Dly/pk$\times 10^{-1}$
.3	72.9	0.0471	1004.6	0.0166
.4	93.6	0.1305	1004.6	0.0468
.5	125.4	0.2981	1004.6	0.1066
.6	175.9	0.6322	1004.6	0.2349
.7	242.0	1.2105	1004.6	0.4694
.8	333.1	2.3134	1004.6	0.9653
.9	487.9	5.0856	1004.6	2.2323

TABLE I

AVERAGE ENERGY/PACKET AND AVERAGE QUEUING DELAY/PACKET FOR LAZY AND DETERMINISTIC OVER AN INFINITE TIME HORIZON. DELAY VALUES ARE IN MILLISECONDS.

We now compare the specific lazy schedule that sets $\tau_{Lazy}(b) = \alpha\frac{(1+b)}{\lambda_{max}}(\frac{\pi^2}{6} - \sum_{k=1}^{b}\frac{1}{k^2})$ to a deterministic schedule with $\tau_{Det}(b) = \frac{\alpha}{\lambda_{max}}$. Note that as long as $\alpha < 1$, both scheduling algorithms ensure stability for arrival rates less than λ_{max}. We performed simulations using both scheduling algorithms for $\alpha = .95$, $\lambda_{max} = 1$, varying λ from .3 to .9. To allow energy and delay to come close to equilibrium, each simulation was performed for $50,000$ arrivals. The results are given in Table I.

The energy/packet values in Table I are dimensionless due to the normalization with noise PSD (see Equation (15)), and the energy values correspond to average SNR per packet of approximately 22 dB to 36 dB for Lazy, and 36 dB for Deterministic.

The results in the table demonstrate that the lazy schedule achieves significantly lower energy than the deteministic with moderate increase in average delay. This energy saving is significant since for a given mean service time, the deterministic schedule achieves the smallest average delay among all schedules that are independent of the arrival process and hence oblivious to backlogs [17]. In turn this implies that the deterministic schedule has the largest transmission times and hence the lowest energy among backlog-oblivious schedules. The fact that our suboptimal lazy schedule is more energy efficient than the deterministic schedule demonstrates the advantage of lazy scheduling.

V. CONCLUSIONS

Conservation of energy is a key concern in the design of wireless networks. Most of the research to date has focused on transmission power control schemes for interference mitigation and only indirectly address energy conservation. In this paper we put forth the idea of conserving energy by lazy scheduling of packet transmissions. This is motivated by the observation that in many channel coding schemes, the energy required to transmit a packet over a wireless link can be significantly reduced by lowering transmission power and transmitting the packet over a longer period of time. However, information is often time-critical or delay-sensitive, and transmission schedules cannot be made too lazy by letting transmission times be arbitrarily long. We therefore considered packet transmission schedules that minimize energy subject to a deadline or a delay constraint. Specifically, we obtained an optimal offline schedule for a node operating under a deadline constraint. An inspection of the form of this schedule naturally lead us to an online schedule, which was shown, through simulations, to be quite energy-efficient. Finally, we relaxed the deadline constraint and provided an exact probabilistic analysis of our offline scheduling algorithm. We then devised an online algorithm, which varies transmission times according to backlog and showed that it is more energy efficient than a deterministic schedule with the same stability region and similar delay.

Several important problems remain open. The most obvious is that of finding the optimal online schedule in the finite and infinite T cases. The question of how much energy can be saved by using lazy scheduling in practice has not been addressed in the paper. The theoretical and simulation results we presented, however, are encouraging enough to warrant further investigation into the use of lazy scheduling in real world wireless networks.

Acknowledgements: The authors thank Chandra Nair for his help with some of the simulations.

REFERENCES

[1] N. Bambos, "Towards Power-Sensitive Network Architectures in Wireless Communications", *IEEE Personal Communications*, 5(3), 1998.

[2] N. Bambos and S. Kandukuri, "Power Control Multiple Access (PCMA) in Wireless Networks", 1999.

[3] N. Bambos and G. Pottie, "Power control based admission policies in cellular radio networks", *GLOBECOM '92*

[4] E. Uysal Biyikoglu, B. Prabhakar and A. El Gamal, "Energy-Efficient Transmission over a Wireless Link via Lazy Packet Scheduling," *CSL Technical Rep. CSL-TR-00-810*, Stanford University, December 2000.

[5] A. Chockalingam, M. Zorzi, "Energy Efficiency of Media Access Protocols for Mobile Data Networks," *IEEE Trans. Comm.*, 46, pp. 1418-21, 1998.

[6] T. Cover, J. Thomas, *Elements of Information Theory*, Wiley Series in Telecommunications, John Wiley & Sons, 1991.

[7] R. Durrett, *Probability:Theory and Examples*, Duxbury Press, 1996.

[8] G.J. Foschini, "A Simple Distributed Autonomous Power Control Algorithm and its Convergence", *IEEE Trans. on Vehicular Technology*, 42(4), 1993.

[9] R. G. Gallager, *Discrete Stochastic Processes*, Kluwer Academic Publishers, Boston, 1995.

[10] A. Goldsmith, "Capacity and dynamic resource allocation in broadcast fading channels," 33^{th} *Annu. Allerton Conf. Communication, Control and Computing*, pp. 915-924.

[11] S. Grandhi, J. Zander and R. Yates, "Constrained power control", *International Journal of Wireless Personal Communications*, 1(4), 1995.

[12] S. Hanly and D. Tse, "Power Control and Capacity of Spread-spectrum Wireless Networks", *Automatica*, 35(12), 1999.

[13] D. Mitra, "An asynchronous distributed algorithm for power control in cellular radio systems", *Proc 4^{th} WINLAB Workshop*, 1993.

[14] G.J. Pottie, "Wireless sensor networks", *Information Theory Workshop '98*, Killarney, Ireland. June 1998.
[15] J.G. Proakis and M. Salehi, *Communication Systems Engineering*, Prentice-Hall, 1994.
[16] P. Viswanath, V. Anantharam and D. Tse, "Optimal Sequences, Power Control and Capacity of Synchronous CDMA Systems with Linear MMSE Multiuser Receivers", IEEE Transactions on Information Theory, vol. 45(6), Sept., 1999.
[17] J. Walrand, *An Introduction to Queueing Networks*, Prentice Hall, 1988.
[18] J. Zander, "Transmitter Power Control for Co-channel Interference Management in Cellular Radio Systems", *Proc 4th WINLAB Workshop*, 1993.

VI. Appendix

Consider a transmitter which, at time 0, has b packets in the queue. Suppose that M packets arrive at this node in $[0,T)$, with the first of these arriving at time 0. This situation can be modeled as $M+b$ packets arriving in $[0,T)$ with $d_1 = \cdots = d_b = 0$ and $\sum_{i=1}^{M+b} d_i = T$. Then, as we have seen in Section IV-A, the optimal offline schedule will transmit the first packet for an amount of time, say $\tau_M(b)$, which is given by

$$\tau_M(b) = \max_{k\in\{1,\ldots,M+b\}} \{\frac{1}{k}\sum_{i=i}^{k} d_i\} \quad (16)$$

$$= \max_{k\in\{1,\ldots,M\}} \{\frac{1}{k+b}\sum_{i=b}^{k} d_i\}. \quad (17)$$

Here we analyze the optimal offline schedule by allowing T to approach infinity. Thus suppose that the arrivals in $[0,T)$ occur as a rate λ Poisson process and let T go to infinity to get

$$\tau(b) = \sup_{\{k\geq 1\}} \{\frac{1}{k+b}\sum_{i=1}^{k} D_i\}, \quad (18)$$

where the D_i are i.i.d. mean $1/\lambda$ exponential random variables. Let $S_i = \sum_{j=1}^{i} D_i$, and let

$$\tau_n(b) = \max_{\{1\leq i\leq n\}} \{\frac{1}{i+b} S_i\}.$$

Lemma 3:

$$E(\tau_n(b)) = \frac{1+b}{\lambda}\sum_{k=1}^{n}\frac{1}{(k+b)^2} \quad (19)$$

Proof: The following recursion can be shown to hold using an induction argument for $n \geq 2$:

$$E(\tau_n(b)) = E(\tau_{n-1}(b)) + \frac{(1+b)}{\lambda(n+b)^2} \quad (20)$$

Since $E(\tau_1(b)) = \frac{1}{\lambda(1+b)}$, the above recursion implies the lemma. Details can be found in [4]. ■

Corollary 1: For $\tau(b) = \sup_{i\geq 1}\frac{S_i}{i} = \lim_{n\to\infty}\tau_n(b)$

$$E(\tau(b)) = \frac{(1+b)}{\lambda}(\frac{\pi^2}{6} - \sum_{m=1}^{b}\frac{1}{m^2}) \quad (21)$$

Proof: Since $\tau_n(b)$ increases in n, by the Monotone Convergence Theorem [7], we obtain that

$$E(\tau(b)) = \lim_{n\to\infty}\frac{1+b}{\lambda}\sum_{k=1}^{n}\frac{1}{(k+b)^2}$$

$$= \frac{(1+b)}{\lambda}(\frac{\pi^2}{6} - \sum_{m=1}^{b}\frac{1}{m^2}).$$

■