This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2007 proceedings.

Energy Efficient Scheduling with Individual Packet Delay Constraints: Offline and Online Results

Wanshi Chen
Qualcomm Inc.
San Diego, CA 92121
wanshic@qualcomm.com

Michael J. Neely and Urbashi Mitra
Dept. of Electrical Engineering
University of Southern California
Los Angeles, USA
{mjneely, ubli}@usc.edu

***Abstract*— This paper focuses on energy-efficient packet transmission with individual packet delay constraints. The optimal offline scheduler (vis-à-vis total transmission energy), assuming information of all packet arrivals before scheduling, was developed by Zafer, *et al.* (2005) and Chen *et al.* (2006). This paper shows that when packet inter-arrival times are identically and independently distributed (*i.i.d.*), the resulting optimal transmission durations of packets m and $M-m+1$, $m \in [1,\cdots,M], M \geq 1$, are identically distributed. This symmetry property leads to a simple and exact solution of the average packet delay under the optimal offline schedule. Two heuristic online scheduling algorithms, which assume no future arrival information, are then studied. These online schedulers are compared with the optimal offline scheduler in terms of delay and energy performance via analysis and simulations. While both online schedulers are inherently inferior, one online scheduler is shown to achieve a comparable energy performance to the optimal offline scheduler in a wide range of scenarios.**

I. Introduction

Future wireless applications are anticipated to require high-speed data links in a limited frequency bandwidth. Various quality-of-service (QoS) requirements, *e.g.*, delay constraints imposed by applications such as voice over Internet protocol (VoIP), challenge current system designs. For many scenarios of interest, the energy required for packet transmission is a monotonically decreasing function of the transmission duration and often convex [1]. Thus, by tolerating more delay, significantly less energy is required when other conditions, such as target packet transmission quality, remain unchanged. The fundamental tradeoff between packet delay and transmission rate has been extensively studied for various scenarios: over a single link [1][2][3][4][5][6], in *ad hoc* networks [7][8][9][10][11], and in cellular networks [12][13][14].

In [1], the optimal energy-efficient scheduling algorithm in minimizing the total transmission energy was developed for a group of packets subject to a *single transmission deadline*. This optimal algorithm assumed knowledge of the total number of packets and the inter-arrival times of these packets before packet scheduling. As a result, it is an *offline* scheduling algorithm. An *online* algorithm, which assumed information of the current scheduling backlog and a maximum packet arrival rate, was also developed in [1]. Online scheduling for the single deadline model was also treated in [6] in which a stochastic optimal control algorithm was developed. Note that since all packets observe a single transmission deadline, the scheduling algorithm may result in large per packet delays, especially when the total number of packets to be transmitted is very large. In fact, it is shown in [15], that under a Poisson arrival model, the average delay associated with the optimal offline scheduler in [1] grows monotonically and at a rate close to $\sqrt{M}$, where M is the total number of data packets. Such potentially significant *individual packet delay* is not desirable especially for delay-sensitive applications and for practical implementation.

Energy-efficient transmission with *individual* packet delay constraints were studied in [5][6][15][16]. Dynamic programming was adopted in [16] to search for optimal online scheduling under dynamic channels. It was proven in [5] that all online scheduling can be expressed as a time-varying low-pass linear filter. Upper and lower bounds on optimal online scheduling were presented, and a 'water-filling' rule was proposed to schedule packets in a slotted system. Offline scheduling with general arrivals and QoS constraints was considered in [6], where an optimal scheduling procedure was developed. A special case of optimally flushing a static buffer with individual delay constraints was also presented in [6]. A similar problem was treated in [15] for optimal offline scheduling with dynamic arrivals and individual delay constraints.

In this paper, we first analyze the properties of the optimal offline scheduler for the *individual delay constraint model*. It is shown that when packet inter-arrival times are identically and independently (*i.i.d.*) distributed, the optimal transmission durations of packet m and packet $M-m+1$, $m \in [1,\cdots,M], M \geq 1$ are also identically distributed. In other words, the optimal transmission duration vector exhibits a symmetric property. This important property makes it possible to obtain a simple and exact solution of the average packet delay (including queuing and transmission delays) for any *i.i.d.* inter-arrival times under the optimal scheduling. In fact, when M is large, the expression for the average packet delay (including queuing and transmission delays) converges to $(T+E[\min\{d,T\}])/2$, where T is the delay constraint and d is a random variable with the same distribution as the packet inter-arrival time.

We then study two heuristic online schedulers for the *individual delay constraint model*. The first online scheduler generalizes the optimal static buffer flushing algorithm in [6] to a system with dynamic packet arrivals and departures.

The conditions under which the optimal buffer flushing rate needs to be updated are identified and characterized. The second online scheduler is a simple frame-based iterative buffering and scheduling scheme, loosely linked to the iterative minimum emptying time (IMET) algorithm in [17]. For each frame duration of $T/2$, where T is the individual delay constraint, the scheduler transmits packets, if any, buffered during the previous frame with the same transmission duration. These two online schedulers are compared with the optimal offline scheduling algorithms in terms of packet delay and transmission energy performance via analysis and simulations. While both online schedulers are inherently inferior, the first online scheduler is shown to achieve a comparable energy performance to the optimal offline scheduler in a wide range of scenarios.

This paper is organized as follows. In Section II, the system model is described. The optimal offline scheduling algorithm for the individual delay constraint model is described in Section III. The symmetry property of the optimal offline scheduling and the resulting delay performance are analyzed in Section IV. The online schedulers and the corresponding delay performance are investigated in Section V. Numerical results are given in Section VI. Finally, some concluding remarks are drawn in Section VII. The Appendix extends the optimal offline scheduling to the scenario of unequal packet sizes and unequal individual delay constraints.

II. System Model

Suppose there are M packets to be transmitted through an additive white Gaussian noise (AWGN) channel, with packet arrival times $t_i, i = 1, \cdots, M$. Without loss of generality, the arrival time of the first packet is assumed to be 0, *i.e.*, $t_1 = 0$. The packet arrivals are assumed to be random, following a known distribution function. Each packet has its own total delay constraint, denoted by T_i, such that by the deadline $t_i + T_i$, the packet has to be completely delivered. This is illustrated in Fig. 1. The packet size is assumed to have B_i bits, $i = 1, \cdots, M$. Note that the individual delay constraints T_i are not necessarily the same for all packets. Similarly, different packets may have different sizes. The packet inter-arrival times, denoted by $d_i = t_{i+1} - t_i, i = 1, \cdots, M$ [1], are random variables.

Each packet will be delivered over the channel with the transmission duration denoted by τ_i. Obviously, τ_i has to satisfy $0 < \tau_i \leq T_i$. The transmission durations of the M packets are denoted by a vector $\vec{\tau} = [\tau_1, \tau_2, \cdots, \tau_M]$. Let $w(\tau)$ be the energy required to transmit a packet with a transmission duration τ. The goal is to find the optimal transmission vector $\vec{\tau}$ such that the total transmission energy of the M packets $w(\vec{\tau}) = \sum_{i=1}^{M} w(\tau_i)$, or equivalently, the average transmission energy $w(\vec{\tau})/M$, is minimized subject to the satisfaction of individual packet delay constraints. Similar to [1], we assume the following:

[1] Although there is no so-called inter-arrival time for the last packet, for convenience of presentation, we still call $d_M = T_M$ the inter-arrival time of packet M.

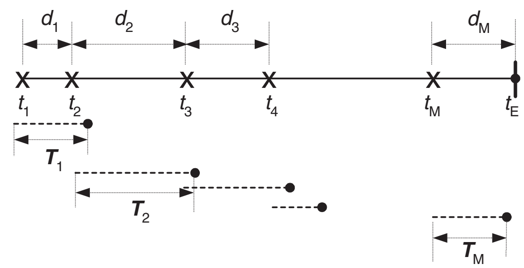

Fig. 1. The individual delay constraint model.

- $w(\tau)$ is non-negative
- $w(\tau)$ is monotonically decreasing in τ
- $w(\tau)$ is strictly convex in τ.

The above assumptions hold over a wireless link for many scenarios of interest, as shown in [1].

The *offline* scheduling algorithm for $\vec{\tau}$ is assumed to have knowledge of the inter-arrival time d_i (hence the packet arrival times), and the individual packet delay constraint $T_i, i \in [1, \cdots, M]$. The *online* schedulers, on the other hand, only assume information of the current scheduling backlog. As in [1], all schedulers are assumed to follow the first-in-first-out (FIFO) service rule, the causality constraint, and the non-idling condition whenever feasible. The FIFO rule means that packets are transmitted in the same order they arrive. The causality constraint ensures that a packet cannot be transmitted before it arrives. The non-idling condition comes from the assumption that the energy function is a decreasing function of the transmission duration τ, and subsequently, the total transmission energy $w(\vec{\tau})$ can always be reduced by increasing the transmission duration for one or more packets. Note that, however, different from the single transmission deadline model, idling periods may be inevitable for any feasible schedulers for the individual delay constraint model. That is, when the inter-arrival time $d_i > T_i$, $i \in [1, \cdots, M]$, the amount of $d_i - T_i > 0$ time resource can not be utilized and an idling period becomes necessary. This issue will be discussed in more detail in the next section.

In the sequel, unless explicitly specified, we will focus on *equal individual delay constraints*, *i.e.*, $T = T_1 = \cdots = T_M$, and *equal packet sizes*, *i.e.*, $B = B_1 = \cdots = B_M$. Most of the results presented herein can be extended to the unequal scenarios.

III. Optimal Offline Schedule

We now summarize the optimal offline scheduling algorithm for the individual delay constraint model [15][6].

First, as discussed in Section II, an idle period is inevitable when $d_i > T$, as the amount $d_i - T > 0$ of time resource can not be utilized by the scheduler. These inevitable idlings can be eliminated if d_i is upper bounded by T, *i.e.*, by using $\vec{\hat{d}} = \min\{\vec{d}, T\}$. On the other hand, when $d_i \geq T$, the i-th packet has to be delivered before (if $d_i > T$) or at (if $d_i = T$) the time when the next packet $i+1$ arrives. As a result, the scheduling of packets with indices $\leq i$ and the scheduling packets with indices $\geq i+1$ are independent. Note that packets $\leq i$ are not affected by replacing the inter-arrival time of the last packet, *i.e.*, d_i, by $T = \hat{d}_i$. Packets $\geq i+1$ are not affected by the replacement as well because the $(i+1)$-th packet is treated as if it were the first arrival anyway. Thus, the optimal schedule with inter-arrival time vector $\vec{\hat{d}}$ would yield exactly the same transmission duration vector $\vec{\tau}$ as with $\vec{d}$. In addition, the resulting packet delay performance would not change if $\vec{\hat{d}}$ replaces $\vec{d}$ in the optimal scheduling.

Therefore, in the following, we will focus on using $\vec{\hat{d}}$ for the optimal schedule. Note that, in contrast to $\vec{d}$, scheduling with $\vec{\hat{d}}$ for the individual delay constraint model strictly satisfies the non-idling condition as in the single deadline model [1]. Thus, we obtain the following feasible scheduling constraints:

$$\begin{array}{l} (i): \left\{ \begin{array}{l} \sum_{i=1}^{k} \tau_i \geq \sum_{i=1}^{k} \hat{d}_i, k \in [1, \cdots, M-1], \text{ and} \\ \sum_{i=1}^{M} \tau_i = \sum_{i=1}^{M} \hat{d}_i, \end{array} \right. \\ (ii): q_k = \sum_{i=1}^{k} \tau_i - \sum_{i=1}^{k-1} \hat{d}_i \leq T, k \in [1, \cdots, M], \end{array} \tag{1}$$

where $\sum_{i=1}^{k} \tau_i$ is the departure time of packet k, while $\sum_{i=1}^{k} \hat{d}_i$ is the arrival time of packet $k+1$ (recall the assumption that the first packet arrives at time 0). The first inequality in (i) indicates that when packet $k+1$ arrives, packet k either is still in transmission or just finished its transmission. The expression $\sum_{i=1}^{M} \tau_i = \sum_{i=1}^{M} \hat{d}_i$ means back-to-back transmissions with no idling in between. The variable q_k in (ii) denotes the delay for packet k, defined as the difference between the packet departure time (completed packet delivery) and the packet arrival time.

From the recursive optimal offline scheduling algorithms [15][6] and using $\vec{\hat{d}}$, we can obtain a closed form expression of the optimal transmission duration of the first packet as

$$\tau_1 = \max_{1 \leq i \leq M} \tau_{1[i]}, \tag{2}$$

where $\tau_{1[i]}, i \geq 1$ is given by

$$\tau_{1[i]} = \min \left\{ \frac{\sum_{m=1}^{i} \hat{d}_m}{i}, T, \frac{\hat{d}_1 + T}{2}, \cdots, \frac{\sum_{m=1}^{i-2} \hat{d}_m + T}{i-1} \right\}. \tag{3}$$

The nested min and max structure in τ_1 reflects the optimal exploitation of both future arrivals and individual delay constraints.

The same criterion in (2) can be used to obtain the transmission durations of any subsequent packets $m > 1$, based on the inter-arrival times of the remaining packets and the queuing delay experienced by packet m before being scheduled for transmission, denoted by $\tilde{q}_m \geq 0$, which is given by

$$\tilde{q}_m = \left\{ \begin{array}{ll} 0 & \text{if } m = 1, \\ \sum_{i=1}^{m-1} (\tau_i - \hat{d}_i) \geq 0 & \text{if } m \in [2, \cdots, M]. \end{array} \right. \tag{4}$$

Note that the delay-constraint factor for packets $m > 1$ now has to be modified by considering the queuing delay $\tilde{q}_m$. That is, instead of T, $T - \tilde{q}_m$ should be used in (3), which yields

$$\tau_{1[i]} = \min \left\{ \frac{-\tilde{q}_m + \sum_{j=m}^{i} \hat{d}_j}{i-m+1}, (T - \tilde{q}_m), \frac{(T-\tilde{q}_m) + \hat{d}_m}{2}, \cdots, \frac{(T-\tilde{q}_m) + \sum_{j=m}^{i-2} \hat{d}_j}{i-m} \right\}. \tag{5}$$

Note that, given a realization of inter-arrival times, the optimal offline scheduling algorithm is unique [6][15].

The extension of the optimal scheduling algorithm to the scenarios of unequal packet sizes and unequal individual delay constraints can be found in the Appendix.

IV. Properties of the Optimal Offline Scheduling

It would be desirable to be able to analytically show the total transmission energy resulting from the optimal offline schedule given a specific set of arrivals and an energy-rate function. However, this appears to be intractable. Thus, we will rely on numerical results to show the transmission energy performance of the optimal offline schedule. However, we are able to analytically derive the average packet delay performance. Herein, we will first analyze the properties of the optimal transmission duration vector by presenting the symmetry property, followed by a subsequent simple and exact solution of the average packet delay performance of the optimal offline scheduling.

A. The Symmetry Property

The following theorem summarizes the symmetry property.

Theorem 4.1: For any $M \geq 1$, when the inter-arrival times $d_m, 1 \leq m \leq M-1$, are *i.i.d.*[2], under the optimal offline scheduling, the optimal transmission durations τ_m and τ_{M-m+1} are identically distributed. In particular, $E\{\tau_m\} = E\{\tau_{M-m+1}\}$, where $E\{.\}$ denotes expectation.

The proof essentially relies on a *time reversal* argument, where we compare a sample path trajectory of the forward running system to a corresponding time reversed system. Consider the original forward system with the individual delay constraint T and a realization of the inter-arrival time vector

$$\vec{d}^{(f)} = [d_1, \cdots, d_{M-1}], \tag{6}$$

where the superscript f denotes the forward system. The absolute arrival times and packet deadlines are thus given by:

$$\begin{array}{rcl} \vec{t}^{(f)}_{arrival} & = & [0, d_1, d_1 + d_2, \cdots, d_1 + d_2 + \ldots + d_{M-1}], \\ \vec{t}^{(f)}_{deadline} & = & \vec{t}^{(f)}_{arrival} + [T, T, \cdots, T]. \end{array}$$

Now define the *reversed system* as a system with the inter-arrival time vector:

$$\vec{d}^{(r)} = [d_{M-1}, d_{M-2}, \cdots, d_2, d_1], \tag{7}$$

where the superscript r denotes the reversed system. This system can be visualized by relating the arrival time of packet k in the reversed system to the *deadline* of packet $M+1-k$ in the forward system. Likewise, the deadline of packet k in

[2]The result also holds for any inter-arrival times where $[d_1, \cdots, d_M]$ and $[d_M, \cdots, d_1]$ are identically distributed. However, for practical interest, we will only focus on the *i.i.d.* inter-arrivals in this paper.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2007 proceedings.

the reversed system corresponds to the *arrival time* of packet $M+1-k$ in the forward system. Specifically, the reversed system has arrival and deadline times given by:

$$t^{(r)}_{arrival,k} = V - t^{(f)}_{deadline,M+1-k}, \forall k \in [1, \cdots, M], \quad (8)$$
$$t^{(r)}_{deadline,k} = V - t^{(f)}_{arrival,M+1-k}, \forall k \in [1, \cdots, M], \quad (9)$$

where $V \triangleq d_1 + d_2 + \ldots + d_{M-1} + T$ is the total time duration of the scheduling algorithm in the forward running system. Thus, the first arrival of the reversed system is indeed at time 0 (corresponding to time V on the time axis of the forward system).

The optimal offline scheduling algorithm can be viewed as a function of an $M-1$ dimensional inter-arrival time vector $\vec{d} = [d_1, d_2, \cdots, d_{M-1}]$ to a unique M dimensional service time vector $\vec{\tau} = [\tau_1, \tau_2, \cdots, \tau_M]$. Let $\vec{\Phi}(\vec{d})$ represent this function (for a given individual deadline constraint T). Thus, $\vec{\tau} = \vec{\Phi}(\vec{d})$ is the unique optimal transmission duration vector for a given inter-arrival vector $\vec{d}$, where $\tau_k = \Phi_k(\vec{d})$ for $k \in [1, \cdots, M]$. Note that $\vec{\Phi}(\vec{d})$ is measurable, as the optimal algorithm in Section III involves only simple summation, multiplication, and max/min operations on the $\vec{d}$ vector.

Now, we are ready to prove Theorem 4.1:

Proof: (Theorem 4.1) The theorem follows from the following two claims:

Claim 1: $\vec{\Phi}(\vec{d}^{(f)})$ and $\vec{\Phi}(\vec{d}^{(r)})$ are identically distributed. Consequently, if $\vec{\tau}^{(f)} = \vec{\Phi}(\vec{d}^{(f)})$ and $\vec{\tau}^{(r)} = \vec{\Phi}(\vec{d}^{(r)})$, then $\tau^{(f)}_k$ and $\tau^{(r)}_k$ are identically distributed for any $k \in [1, \cdots, M]$.

Claim 2: For any particular inter-arrival time vector $\vec{d}^{(f)}$, if $\vec{\tau}^{(f)} = [\tau_1, \cdots, \tau_M]$ is the unique optimal transmission time vector for the forward system, $\vec{\tau}^{(r)} = [\tau_M, \cdots, \tau_1]$ is the unique optimal transmission time vector for the reversed system.

The proof of Claim 1 follows directly from the fact that since $\vec{d}^{(f)}$ and $\vec{d}^{(r)}$ are identically distributed, any function of them are identically distributed as well.

To prove Claim 2, let $\vec{\tau}^{(f)} = [\tau_1, \cdots, \tau_M]$ be the optimal transmission duration vector for the forward system, yielding total energy expenditure $e^{(f)}_{opt} = \sum_{k=1}^{M} w(\tau_k)$. For each packet $k \in [1, \cdots, M]$, let $t^{(f)}_{arrive,k}, t^{(f)}_{start,k}, t^{(f)}_{end,k}$, and $t^{(f)}_{deadline,k}$ represent the time packet k arrives, begins its transmission, ends its transmission, and reaches its deadline, respectively, under the optimal scheduling of the forward system. Clearly, we have:

$$t^{(f)}_{arrive,k} \le t^{(f)}_{start,k} \le t^{(f)}_{end,k} \le t^{(f)}_{deadline,k}.$$

Note that the reversed system can *emulate* the same transmission durations of the forward system, in the following sense: For each packet k in the forward running system, schedule packet $M+1-k$ in the backward running system according to times $t^{(r)}_{arrive,M+1-k}, t^{(r)}_{start,M+1-k}, t^{(r)}_{end,M+1-k}$,

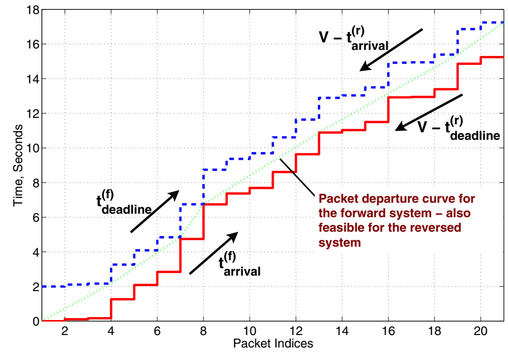

Fig. 2. Illustration of the forward system and the reversed system. The packet departure curve in the forward system, due to the optimal transmission duration vector $\vec{\tau}$, is also feasible and optimal in the reversed system.

and $t^{(r)}_{deadline,M+1-k}$, where:

$$\begin{aligned} t^{(r)}_{arrive,M+1-k} &\triangleq V - t^{(f)}_{deadline,k}, \\ t^{(r)}_{start,M+1-k} &\triangleq V - t^{(f)}_{end,k}, \\ t^{(r)}_{end,M+1-k} &\triangleq V - t^{(f)}_{start,k}, \\ t^{(r)}_{deadline,M+1-k} &\triangleq V - t^{(f)}_{arrive,k}. \end{aligned} \quad (10)$$

Note that this scheduling policy transmits no more than one packet at any time, and that it satisfies the feasibility constraints of the reversed system. Specifically, the causality constraint of the forward running system implies packet k cannot begin its service until it has arrived, so that $t^{(f)}_{arrive,k} \le t^{(f)}_{start,k}$. It follows that:

$$V - t^{(f)}_{arrive,k} \ge V - t^{(f)}_{start,k}. \quad (11)$$

Applying the definitions of $t^{(r)}_{end,M+1-k}$ and $t^{(r)}_{deadline,M+1-k}$ in (10) to the inequality by (11) yields

$$t^{(r)}_{deadline,M+1-k} \ge t^{(r)}_{end,M+1-k}.$$

This ensures that packet $M+1-k$ of the *reversed system* ends its service time on or before its deadline constraint. Likewise, the deadline constraint $t^{(f)}_{end,k} \le t^{(f)}_{deadline,k}$ for packet k under the forward system implies the *causality constraint* for packet $M+1-k$ in the reversed system. This is illustrated in Figure 2.

Further note that this policy schedules packet $M+1-k$ in the reversed system for a service time exactly equal to τ_k, the service time of packet k in the forward system. It follows that this emulation achieves exactly the same energy expenditure $e^{(f)}_{opt}$ as the forward system. Therefore, the *optimal* energy expenditure of the reversed system is less than or equal to that of the forward system. However, because the forward system can be viewed as a 'reversed' reversed system, it likewise follows that the optimal energy expenditure of the forward system is less than or equal to that of the reversed system. Hence, both the forward and reversed systems have exactly the same optimal energy expenditure, with (unique) optimal transmission duration vectors given by $[\tau_1, \cdots, \tau_M]$ and $[\tau_M, \cdots, \tau_1]$, respectively.

The two claims above immediately imply the theorem because, for any $k \in [1, \cdots, M]$, we have that $\tau_k^{(f)}$ is identically distributed to $\tau_k^{(r)}$ (Claim 1), and $\tau_k^{(r)} = \tau_{M+1-k}^{(f)}$ (Claim 2). Thus, $\tau_k^{(f)}$ and $\tau_{M+1-k}^{(f)}$ are identically distributed. ■

Note that for the *single transmission deadline model* [1], the optimal transmission duration vector exhibits a monotonically non-increasing property for *each realization* of inter-arrival times. This is different from the *statistically* symmetric property for the *individual delay constraint model* discussed here.

B. Packet Delay Performance

Now we will utilize the symmetry property of the optimal $\vec{\tau}$ to obtain a simple and exact solution of the average packet delay performance for the optimal offline scheduler. First, define the average packet delay as:

$$\bar{q} \triangleq E\{\frac{1}{M}\sum_{m=1}^{M} q_m\}$$

where q_m is the delay experienced by packet m under the optimal offline schedule with a particular realization of the inter-arrival time vector (see (1)), and the expectation is taken over all realizations of packet inter-arrival times. We have:

Lemma 4.2: For any $M \geq 1$, when the inter-arrival times $d_m, 1 \leq m \leq M-1$, are *i.i.d.*, under the optimal offline schedule, the average packet delay is given by

$$\bar{q} = \bar{\hat{d}} + \frac{M+1}{2M}(T - \bar{\hat{d}}). \tag{12}$$

where $\bar{\hat{d}} = E\{\hat{d}_m\}$ and $\hat{d}_m = \min\{d_m, T\}$. When $M \to \infty$, $\bar{q} \to (T + \bar{\hat{d}})/2$.

Proof: First, from Section III, we know $\vec{\hat{d}}$ yields the same optimal transmission duration vector $\vec{\tau}$ as $\vec{d}$, and subsequently, the same packet delay performance. From (1), the delay for packet m is $q_m = \sum_{l=1}^{m} \tau_l - \sum_{l=1}^{m-1} \hat{d}_l$. Thus,

$$\begin{aligned}
\bar{q} &\triangleq \tfrac{1}{M}\textstyle\sum_{m=1}^{M} E\{q_m\} \\
&= \tfrac{1}{M}\textstyle\sum_{m=1}^{M}\left[\sum_{l=1}^{m} E\{\tau_l\} - \sum_{l=1}^{m-1} E\{\hat{d}_l\}\right] \\
&\overset{(a)}{=} \tfrac{1}{M}\textstyle\sum_{m=1}^{M}(M-m+1)E\{\tau_m\} - \tfrac{1}{M}\sum_{m=1}^{M}(m-1)\bar{\hat{d}} \\
&\overset{(b)}{=} \tfrac{1}{M}\textstyle\sum_{m=1}^{M}\tfrac{M+1}{2}E\{\tau_m\} - \tfrac{1}{M}\tfrac{M(M-1)}{2}\bar{\hat{d}} \\
&\overset{(c)}{=} \tfrac{M+1}{2M}[(M-1)\bar{\hat{d}} + T] - \tfrac{M-1}{2}\bar{\hat{d}} \\
&= \bar{\hat{d}} + \tfrac{M+1}{2M}(T - \bar{\hat{d}}),
\end{aligned}$$

where (a) holds by counting the number of occurrences of each item, the first term of (b) comes from the symmetry property $E\{\tau_m\} = E\{\tau_{M-m+1}\}, \forall m \in [1, \cdots, M]$, and equivalently, there are $(M+1)/2$ copies of each $E\{\tau_m\}$, and the first term of (c) is due to the fact that $\sum_{m=1}^{M} E\{\tau_m\} = (M-1)\bar{\hat{d}} + T$, *i.e.*, the non-idling scheduling. ■

Corollary 4.2.1: The average queuing delay (excluding the transmission time), *i.e.*, $\bar{q} - \frac{1}{M}\sum_{m=1}^{M} E\{\tau_m\}$, is given by

$$\bar{\bar{q}} = \frac{M-1}{2M}(T - \bar{\hat{d}}).$$

When $M \to \infty$, $\bar{\bar{q}} \to (T - \bar{\hat{d}})/2$.

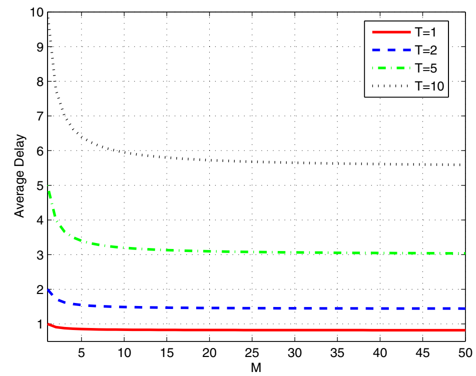

Fig. 3. The average packet delay vs. M for the individual delay constraint model, $\lambda = 1$.

Proof: This comes directly from Lemma 4.2 and the fact that $\sum_{m=1}^{M} E\{\tau_m\} = (M-1)\bar{\hat{d}} + T$. ■

Lemma 4.2 indicates that under the *i.i.d.* assumption of $\vec{\hat{d}}$, the average packet delay is roughly half of the delay constraint when M is sufficiently large and $T \gg \bar{\hat{d}}$. This is illustrated in Figure 3 for different individual delay constraints, under a Poisson arrival model of rate λ (thus, $\bar{\hat{d}}$ is equal to $(1 - e^{-\lambda T})/\lambda$). As can be seen, when M increases, the average packet delay decreases and converges to a fixed value. Under typical scenarios, $\bar{\hat{d}}$ is roughly equal to $\bar{d}$ (without being upper-bounded by T), and hence $1/\lambda$. From Little's Theorem [18], the average number of packets is approximately $(\lambda T + 1)/2$ under the optimal scheduling for the individual delay constraint model.

This delay performance is in contrast to the potentially unbounded average packet delay performance of the optimal offline scheduling algorithm for the *single deadline model* [15]. In fact, assuming a Poisson packet arrival of rate λ, the average packet delay $\bar{q}_M^{single}$ for a total number of M packets was approximated by [15]

$$\bar{q}_M^{single} \approx \frac{1}{\lambda}\left(\frac{\sqrt{2\pi}}{6}\sqrt{M} + 1\right). \tag{13}$$

V. Online Schedulers and Their Properties

In this section, we will investigate two online schedulers. The online schedulers only assume information of the current scheduling backlog. The first scheduler extends the optimal static buffer flushing algorithm in [6] to a system with dynamic packet arrivals and departures. The second scheduler is an IMET-like algorithm [17] in which packets are buffered and scheduled on a fixed duration basis.

A. Optimal Buffer Flushing Based Online Scheduling

The optimal algorithm of flushing a static buffer with a finite number of packets of various discrete individual delay constraints was investigated in [6], a re-formulation of the scheduling issue in [5]. Assume at a time instant, there are a finite number (*e.g.*, $K \geq 1$[3]) of discrete individual delay constraints in the buffer, *i.e.*, $T_1 \leq T_2 \leq \cdots \leq T_K$, with packet sizes of $B_1, B_2, \cdots, B_K$, respectively. It is shown in [6] that the optimal transmission duration of the *head packet*, *i.e.*, the packet with the smallest delay constraint, can be written as:

$$\tau_1 = B_1 \min_{k \in [1, \cdots, K]} \frac{T_k}{\sum_{i=1}^{k} B_i}. \tag{14}$$

Here we extend this static optimal buffer-flushing algorithm to a continuous time system with dynamic packet arrivals and departures. However, the transmission duration of the head packet by (14) may no longer hold when a new packet arrives. In other words, the optimal *transmission rate* inherent in (14) may be kept until upon a new packet arrival, at which point a new optimal transmission rate may become necessary. Therefore, we can re-write (14) in terms of the optimal buffer flushing rate as

$$r_{opt} = \max_{k \in [1, \cdots, K]} \frac{\sum_{i=1}^{k} B_i}{T_k}. \tag{15}$$

It is worth noting that although we assume equal packet sizes and equal delay constraints, at a particular time instant, the pending packets in the buffer do not necessarily have the same packet sizes and delay constraints. This is because packets may arrive at different times, and the head packet in the buffer may already in the process of transmission[4].

When a new packet arrives, the new optimal flushing rate is no less than the original rate before the new packet arrival. This is obvious as a new term is added to the right hand side of (15). On the other hand, the optimal rate in (15) may also need to be updated upon a packet departure, as the first entry on the right hand of (15) disappears. If the departed packet has a delay less than T, the optimal flushing rate remains unchanged. However, if its delay is exactly equal to T, the optimal flushing rate may be decreased. This is due to the fact that the departed head packet satisfies $B_1/T_1 = r_{opt}$. To sum up, at any time, the scheduler chooses a transmission rate based on (15), and may

- Increase the rate upon a new packet arrival, or
- Decrease the rate upon a packet departure with a delay exactly equal to T,

where B_i and T_i in (15) are the current *residual* packet size and the *remaining* time to the deadline for packet i.

Lemma 5.1: The online scheduler by (15) guarantees that all packets meet their own delay constraints.

Proof: This can be easily checked by noting the FIFO assumption and the fact that at any time $t \geq 0$, we have $r_{opt} \geq B_1/T_1$. ∎

For convenience, we will denote the above online scheduler as the *online flush scheduler*. It is worth emphasizing that no future arrival information, completely or partially, is assumed in the design of the above online scheduling algorithm. At any given time instant, the online flush scheduler yields the optimal transmission rate when there are no future packet arrivals. In a system of dynamic arrivals, the above online scheduler may no longer be optimal when additional future arrival information is anticipated and incorporated. For instance, a stochastic optimal control algorithm, which anticipates future arrivals, was investigated in [6] for packets subject to *a single transmission deadline*.

Obviously, the online flush scheduler results in a transmission energy no less than that of the optimal offline scheduler. The following Lemma characterizes its delay performance:

Lemma 5.2: Given $M \geq 1$ and any particular inter-arrival time vector $\vec{d}$, for each packet $m \in [1, \cdots, M]$, the packet delay under the online flush scheduler by (15), q_m^{flush}, is no less than that of the optimal offline scheduler by (2), $q_m^{offline}$. In particular, if $q_m^{offline} = T$, then $q_m^{flush} = T$. In addition, $E\{q^{flush}\} \geq E\{q^{offline}\}$.

Proof: Note that due to the FIFO constraint, when both schedulers have the same queue length, they must have the same buffered packets. Now consider at any time $t \geq 0$, whenever the two schedulers have the same buffered packet sizes, reflected by $B_1, B_2, \cdots, B_K$ corresponding to delay constraints $T_1 \leq T_2 \leq \cdots \leq T_K$, respectively. This certainly holds true at time 0 when the first packet arrives, where $B_1 = B$ and $T_1 = T$. At time t^+, the flush scheduler chooses a transmission rate $r^{flush}(t^+)$ by (15), using only current backlog information B_i and T_i, $i \in [1, \cdots, K]$. The optimal offline scheduler, however, in addition to current backlog information, also exploits future arrival information. Thus, at time t^+, the optimal offline scheduler always chooses a rate $r^{offline}(t^+)$ no less than that of the flush scheduler, *i.e.*, $r^{offline}(t^+) \geq r^{flush}(t^+)$. Thus, the unfinished amount of work in the flush scheduler is never less than that of the offline scheduler. ∎

More delay performance comparisons between the optimal offline scheduler and the online flush scheduler will be investigated via simulations in Section VI.

B. The IMET-like Online Scheduler

This online scheduler is loosely linked to the iterative minimum emptying time (IMET) algorithm in [17] and thus is termed as the IMET-like scheduler. The IMET algorithm [17] is a frame-based iterative scheduling algorithm which, given a certain traffic model and channel conditions, determines the minimum required frame duration such that all the packets buffered in the preceding frame can be fully scheduled (without packet dropping) in the current frame. This design idea leads to a simple online scheduler for the individual delay constraint model. In this case, the goal is not to find the minimum frame duration. Instead, given the individual delay constraint T, we can just choose a frame duration of $T/2$, and iteratively buffer and schedule packets on a per $T/2$ basis. In doing this, we can guarantee that all packets will experience

[3]The case when $K = 0$ is trivial.

[4]Note that here a fluid packet departure model is assumed. That is, a transmitted packet is not necessarily an integer number of arrived packets, but may be assembled using fragmented arrival packets up to an arbitrary precision.

a delay no more than T. To be more specific, the IMET-like online scheduler is as follows:

1) Choose the frame duration as $T_f = T/2$.
2) During the first frame, do nothing.
3) For each subsequent frames, ignore all new arrivals during this frame but clear the backlog due to the preceding frame such that each buffered packet is transmitted with the same duration of T_f/N_f, where $N_f \geq 1$ is the number of buffered packets during the previous frame. If $N_f = 0$, do nothing.

Note that unlike the optimal offline scheduler and the online flush scheduler, the IMET-like online scheduler with inter-arrival times $d_i \leq T$, $i \in [1, \ldots, M]$, may still yield idling periods, especially when $d_i > T/2$. These potential idling periods, along with the inefficiency of potentially not fully utilizing the individual delay constraints, makes this simple scheduler inferior to both the optimal offline scheduler and the online flush scheduler, as will be demonstrated in Section VI.

Now let us study the packet delay performance for the IMET-like scheduler. Similar to [1][15], we will focus on a Poisson arrival model such that inter-arrival times follow an exponential distribution. Under the Poisson arrival model, the arrival times of these $N_f \geq 1$ packets in the preceding frame follow a uniform distribution. Under the scheduling scheme, these packets will depart in the current frame with a T_f/N_f increment. Thus, the average delay for these N_f packets is

$$\frac{T_f}{2} + \frac{1}{N_f}\frac{N_f(N_f+1)}{2}\frac{T_f}{N_f} = T_f + \frac{T_f}{2N_f}, N_f \geq 1, \quad (16)$$

where the first term on the left hand side is due to the average packet arrival time in the preceding frame to the boundary of the current frame, while the second term is due to the T_f/N_f increment for a total of N_f packets. Note that when $N_f = 0$, the average delay is zero.

Now, the number of packets arrived in a frame, N_f, follows a Poisson distribution, *i.e.*, $P\{N_f = n\} = e^{-\lambda T_f}(\lambda T_f)^n/n!$, where λT_f is the average number of packets arrived in a frame. Using (16), the average delay of these (on average) λT_f packets can be obtained as

$$\begin{aligned} \bar{q}^{IMET} &= \tfrac{1}{\lambda T_f}\textstyle\sum_{n=1}^{\infty} n e^{-\lambda T_f}\tfrac{(\lambda T_f)^n}{n!}(T_f + \tfrac{T_f}{2n}) \\ &= e^{-\lambda T_f}/\lambda[\textstyle\sum_{n=1}^{\infty}\tfrac{(\lambda T_f)^n}{(n-1)!} + \tfrac{1}{2}\textstyle\sum_{n=1}^{\infty}\tfrac{(\lambda T_f)^n}{n!}] \\ &= e^{-\lambda T_f}/\lambda[\lambda T_f e^{\lambda T_f} + \tfrac{1}{2}(e^{\lambda T_f} - 1)] \\ &= T_f + \tfrac{1}{2\lambda}(1 - e^{-\lambda T_f}) \\ &= \tfrac{T}{2} + \tfrac{1}{2\lambda}(1 - e^{-\lambda T/2}) \end{aligned} \quad (17)$$

where we have used $\bar{q}^{IMET} = E\{nT_f + T_f/2 | n \geq 1\}/(\lambda T_f)$, resulting from renewal theory and the law of large numbers, where the numerator is the average sum of delays of all packets ($n \geq 1$) scheduled over a frame (from (16)), and the denominator is the average number of packets scheduled over a frame duration.

Due to the Poisson arrival assumption in a frame, $\bar{q}^{IMET}$ (see (17)) is not a function of the total number of packets M. However, for a given M, the number of packet arrivals in a frame is finite and upper bounded by M. Thus, the actual average delay for the IMET-like online scheduler is approximately given by (17), and converges to (17) when M approaches infinity.

Note from (12) that, when M is large, the average packet delay of the optimal offline scheduler, $\bar{q}^{offline}$, converges to $\bar{q}^{offline} \rightarrow T/2 + \bar{\hat{d}}/2$, where under the Poisson arrival model, $\bar{\hat{d}} = (1 - e^{-\lambda T})/\lambda$. Compared $\bar{q}^{IMET}$ in (17), it can be seen that when T is reasonably large,

$$\bar{q}^{IMET} \approx \bar{q}^{offline}, \text{ when } M \rightarrow \infty.$$

That is, the IMET-like online scheduler achieves almost the same delay performance as the optimal offline scheduler, while its transmission energy performance may be significantly worse than that of the optimal offline scheduler, as will be shown in the next Section.

VI. Numerical Results

We assume a Poisson arrival rate of $\lambda = 1$ packet/second. The inter-arrival time d_M is fixed at T for both models. The energy function is assumed to be $w(\tau) = \tau(2^{2B/\tau} - 1)$ [1], where B is normalized to be the number of bits per channel use. It is worth noting the optimal transmission duration vector $\vec{\tau}$ and the corresponding average packet delay are *not* a function of B. We will first illustrate the properties of the optimal transmission durations and packet delays, followed by the performance comparison of the optimal offline scheduler and the online schedulers.

A. Properties of the Optimal Offline Scheduling

Figure 4 shows the average optimal transmission durations when $M = 100$. The results were averaged over 10,000 independent simulations. The single transmission deadline model is shown for comparison. As can be seen, the optimal transmission duration vector for the individual delay constraint model exhibits the symmetry property. The minimum transmission duration occurs in the middle of the packets. The single transmission deadline model, on the other hand, exhibits a different property. Its average optimal transmission durations are monotonically decreasing. Note that the average optimal transmission durations of the individual delay constraint model are lower bounded by the average inter-arrival time $\bar{\hat{d}} = (1 - e^{-\lambda T})/\lambda$ (not shown).

Figure 5 shows the average packet delay associated with the optimal transmission durations. It can be seen that the average delay for the individual delay constraint model increases with packet index m, and maximizes at T when $m = M$. The average packet delay, including the transmission delay, is close to $(T + \bar{\hat{d}})/2$ when m is approximately $M/2$. The delay for the single transmission deadline model, however, peaks in the middle of the packets, and is much higher than that of the individual delay constraint model.

B. Comparison Between the Offline and Online Schedulers

Figure 6 shows the normalized transmission energy performance for the two online schedulers when $M = 1000$. The energy normalization is performed with respect to that of

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2007 proceedings.

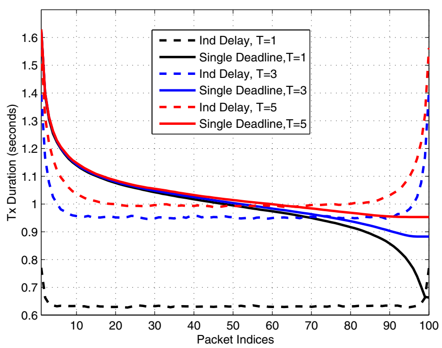

Fig. 4. Average optimal offline transmission durations, $M = 100$.

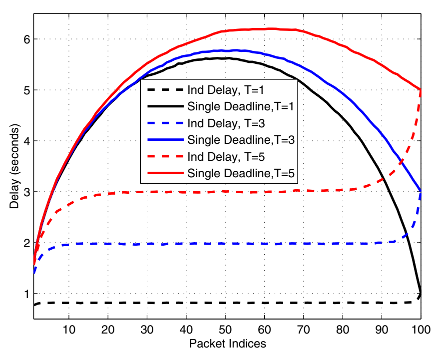

Fig. 5. Average packet delay vs packet indices under the optimal offline schedule, $M = 100$.

the optimal offline scheduler. The results were averaged over 1,000 independent simulations. It can be seen that the *online flush scheduler* achieves a comparable energy performance to the optimal offline scheduler when the normalized packet size is small and/or the individual delay constraint is large. The simple *IMET-like online scheduler*, however, performs significantly worse than both the optimal offline and the online flush schedulers, especially for small delay constraints and large packet sizes.

Figure 7 and Figure 8 show the average packet delay performance for the offline and the online schedulers when $M = 1000$. The online flush scheduler, on average, achieves a larger delay performance than both the two other schedulers, which yields almost the same delay performance. The above delay performance gap grows as the individual delay constraint increases. The average delay for all the three schedulers increases rapidly at the beginning and the end of the set of M packets, and remains flat in the middle. The analytical delay performance results for the optimal offline scheduler in (12) and the IMET-like online scheduler in (17) are also shown in Figure 8 for comparison, and agree very well with the simulations.

VII. Conclusions

In this paper, we investigated the properties of the optimal offline scheduling for packets subject to individual delay

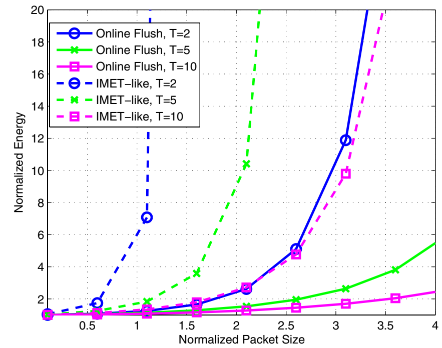

Fig. 6. Average normalized transmission energy for the online schedulers, $M = 1000$.

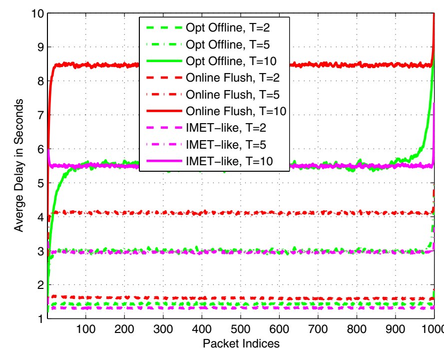

Fig. 7. Average packet delay vs packet indices for the offline and online schedulers, $M = 1000$.

constraints. It was shown that when packet inter-arrival times are identically and independently distributed, the resulting optimal transmission durations exhibit a symmetry property. This important property makes it possible to obtain a simple and exact solution of the average packet delay. Two online schedulers were then studied and compared with the optimal offline scheduling algorithm in terms of packet delay and transmission energy performance via analysis and simulations. While both online schedulers are inherently inferior, the online flush scheduler is shown to achieve a comparable energy performance to the optimal offline scheduler in a wide range of scenarios.

Acknowledgments

This research has been funded in part by one or more of the following grants or organizations: NSF ANI-0087761, NSF Special Projects ANI-0137091, NSF ITR CCF-0313392, NSF Grant OCE-0520324, and the DARPA IT-MANET program, Grant W911NF-07-1-0028.

References

[1] E. Uysal-Biyikoglu, B. Prabhakar, and A. E. Gamal, “Energy-efficient packet transmission over a wireless link,” *IEEE/ACM Trans. Networking*, vol. 10, no. 4, pp. 487–499, Aug 2002.

[2] R. Berry and R. G. Gallager, “Communication over fading channels with delay constraints,” *IEEE Trans. Info. Theory*, vol. 48, no. 5, pp. 1135–1149, May 2002.

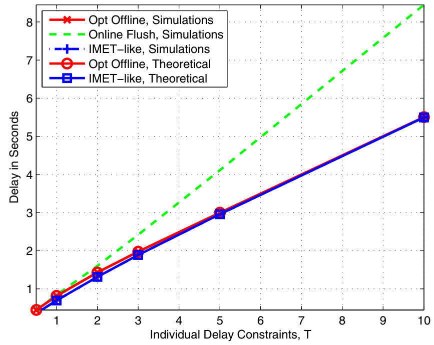

Fig. 8. Average packet delay vs T for the offline and online schedulers, $M = 1000$.

[3] A. Fu, E. Modiano, and J. Tsitsiklis, "Optimal transmission scheduling over a fading channel with energy and deadline constraints," *IEEE Trans. Wireless Communications*, vol. 5, no. 3, pp. 630–641, March 2006.

[4] M. Goyal, A. Kumar, and V. Sharma, "Power constrained and delay optimal policies for scheduling transmission over a fading channel," *IEEE INFOCOM, San Francisco*, pp. 311–320, March 2003.

[5] M. A. Khojastepour and A. Sabharwal, "Delay-constrained scheduling: power efficiency, filter design, and bounds," *IEEE INFOCOM, Hong Kong*, pp. 1939–1950, March 2004.

[6] M. A. Zafer and E. Modiano, "A calculus approach to minimum energy transmission policies with quality of service gurantees," *IEEE INFOCOM, Miami*, pp. 548–559, March 2005.

[7] M. Grossglauser and D. N. C. Tse, "Mobility increases the capacity of ad hoc networks," *IEEE/ACM Trans. Networking*, vol. 10, pp. 477–486, Aug 2002.

[8] M. J. Neely and E. Modiano, "Capacity and delay tradeoffs for ad-hoc mobile networks," *IEEE Trans. on Info. Theory*, vol. 51, pp. 1917–1937, June 2005.

[9] S. Toumpis and A. J. Goldsmith, "Large wireless networks under fading, mobility, and delay constraints," *IEEE INFOCOM, Hong Kong*, pp. 609–619, March 2004.

[10] X. Lin and N. B. Shroff, "Towards achieving the maximum capacity in large mobile wireless networks," *KICS/IEEE Journal of Communications and Networks*, vol. 6, no. 4, pp. 352–361, Dec 2004.

[11] G. Sharma, R. R. Mazumdar, and N. B. Shroff, "Delay and capacity trade-offs in mobile ad hoc networks: a global perspective," *IEEE INFOCOM, Barcelona*, 2006.

[12] C. Wong, C. Tsui, R. Cheng, and K. Letaief, "A real-time sub-carrier allocation scheme for multiple access downlink OFDM transmission," *IEEE 50-th Vehicular Tech. Conference*, vol. 2, pp. 1124–8, 1999.

[13] M. Ferracioli, V. Tralli, and R. Verdone, "Channel based adaptive resource allocation at the MAC layer in UMTS TD-CDMA systems," *IEEE 2000 Vehicular Tech. Conference*, vol. 2, pp. 2549–2555, 2000.

[14] B. Tsybakov, "File transmission over wireless fast fading downlink," *IEEE Trans. Info. Theory*, vol. 48, no. 8, August 2002.

[15] W. Chen and U. Mitra, "Energy efficient scheduling with individual packet delay constraints," *IEEE INFOCOM, Barcelona*, 2006.

[16] D. Rajan, A. Sabharwal, and B. Aazhang, "Delay-bounded packet scheduling of bursty traffic over wireless channels," *IEEE Trans. Info. Theory*, vol. 50, no. 1, pp. 125–144, Jan 2004.

[17] M. J. Neely, J. Sun, and E. Modiano, "Delay and complexity tradeoffs for dynamic routing and power allocation in a wireless network," *Proceedings of the 40th Annual Allerton Conference on Communication, Control, and Computing*, October 2002.

[18] L. Kleinrock, *Queuing Systems Vol. 1: Theory*. Wiley-Interscience, 1975.

Appendix A
Optimal Offline Scheduling for Unequal Packet Sizes and Unequal Individual Delay Constraints

Here we extend the optimal offline schedule for the individual delay constraint model to unequal packet sizes and unequal individual delay constraints. We first start with unequal packet sizes but equal individual delay constraints. The optimal offline schedule for the *single deadline model* with unequal packet sizes was discussed in [1]. Similar to [1], we should now try to equalize per bit packet transmission duration instead of per packet transmission duration. However, we will adopt a derivation approach different from the one in [1]. Indeed, this approach can also be used as an alternative to derive the optimal offline scheduler under unequal packet sizes for the *single deadline model* [1].

Note that the optimal offline scheduler with equal packet sizes and equal individual delay constraints discussed in Section III is valid for any packet inter-arrival times. Now, in case of unequal packet sizes, we can view each *bit* in a packet as a *virtual packet*, and the inter-arrival times between all bits in the same packet are essentially zero. In other words, a packet of B_i bits is treated as if there were B_i equal-size (*i.e.*, one bit) virtual packet arrivals, with virtual inter-arrival times $e_{i,j} = 0, j \in [1, \cdots, B_i - 1]$, and $e_{i,B_i} = \hat{d}_i$. Applying these virtual packet arrivals to (2) and (5), after some derivations, it can be shown the optimal transmission duration for any packet m, under a queuing delay of $\tilde{q}_m$, is given by

$$\tau_m = B_m \max_{m \le i \le M} \tau_{1[i]}, \tag{18}$$

where $\tau_{1[i]}$, $i \in [m, \cdots, M]$, given by

$$\tau_{1[i]} = \min\left\{ \frac{-\tilde{q}_m + \sum_{l=m}^{i} \hat{d}_l}{\sum_{l=m}^{i} B_l}, \frac{(T - \tilde{q}_m)}{B_m}, \frac{(T - \tilde{q}_m) + \hat{d}_m}{B_m + B_{m+1}}, \cdots, \frac{(T - \tilde{q}_m) + \sum_{l=m}^{i-2} \hat{d}_l}{\sum_{l=m}^{i-1} B_l} \right\}. \tag{19}$$

Now when packets have different delay constraints $T_i, i = 1, \cdots, M$, it is natural to assume that the transmission deadline is monotonically non-decreasing, i.e., $t_i + T_i \le t_{i+1} + T_{i+1}$, such that the packets still follow the FIFO rule. Similar scheduling feasibility constraints in (1) still hold, with $\hat{d}_i = \min(d_i, T_i)$, for $1 \le i \le M$. In addition, the check of packet delay constraints should be based on the different individual packet delay constraints T_i, instead of the same constraint T. The optimal offline schedule still follows the exact procedure as discussed in Section III. The optimal transmission duration for packet $m \in [1, \cdots, M]$ can be obtained as

$$\tau_m = B_m \max_{m \le i \le M} \tau_{1[i]}, \tag{20}$$

where $\tau_{1[i]}$, $i \in [m, \cdots, M]$, given by

$$\tau_{1[i]} = \min\left\{ \frac{-\tilde{q}_m + \sum_{l=m}^{i} \hat{d}_l}{\sum_{l=m}^{i} B_l}, \frac{(T_m - \tilde{q}_m)}{B_m}, \frac{(T_{m+1} - \tilde{q}_m) + \hat{d}_m}{B_m + B_{m+1}}, \cdots, \frac{(T_{i-1} - \tilde{q}_m) + \sum_{l=m}^{i-2} \hat{d}_l}{\sum_{l=m}^{i-1} B_l} \right\}. \tag{21}$$

where $\tilde{q}_m \ge 0$ again is the queuing delay of packet m.