Energy Efficient Algorithms for Real-Time Traffic over Fading Wireless Channels

Shuai Zuo, Han Deng, I-Hong Hou

Abstract—**This paper studies the problem of using minimum power to provide satisfactory performance for real-time applications over unreliable and fading wireless channels. We demonstrate that this problem can be formulated as a linear programming problem. However, this formulation involves exponentially many constraints, and many parameters are either unavailable or difficult to compute, which makes it infeasible to employ standard techniques to solve the linear programming problem. Instead, we propose a simple online algorithm for this problem. We prove that our algorithm provides satisfactory performance to all real-time applications, and the total power consumption can be made arbitrarily close to the theoretical lower bound. Further, our algorithm has very low complexity and does not require knowledge of many parameters in the linear programming problem, including the distributions of channel qualities.**

We further extend our algorithm to address systems where real-time applications and non-real-time ones coexist. We demonstrate that our algorithm achieves both low total power consumption and high utility for each non-real-time client while satisfying the performance requirements of real-time clients. Simulation results further provide some insights in setting important parameters of our algorithms, and demonstrate that our algorithm indeed achieves a significant reduction in power consumption.

I. Introduction

Wireless networks have been widely applied to serve real-time applications with stringent per-packet delay. These applications, such as Voice over Internet Protocol (VoIP), interactive multimedia and video streaming, consume more and more wireless resources as the demand increases. On the other hand, with the recent emphasis on green communications, reducing energy consumption while providing satisfactory performance to all real-time applications has become an important concern for future wireless networks.

Since the wireless channels are fading channels [1] and many wireless clients are mobile, it may be infeasible to obtain the clients channel statistics. Therefore, it is vital to develop online algorithms that do not rely on the knowledge of mobility patterns or distributions of channel qualities. Finally, in most scenarios, real-time and non-real-time applications may coexist in the same network, and hence an energy efficient algorithm needs to accommodate the two very different types of applications.

Shuai Zuo is with the CESG and Department of ECE, Texas A&M University, College Station, TX 77843, USA. E-mail: zuosh029@tamu.edu

Han Deng is with the CESG and Department of ECE, Texas A&M University, College Station, TX 77843, USA. E-mail: hdeng@tamu.edu

I-Hong Hou is with the CESG and Department of ECE, Texas A&M University, College Station, TX 77843, USA. E-mail: ihou@tamu.edu

This material is based upon work supported in part by the U. S. Army Research Laboratory and the U. S. Army Research Office under contract/grant number W911NF-15-1-0279 and NPRP Grant 8-1531-2-651 of Qatar National Research Fund (a member of Qatar Foundation). Part of this work has been presented in Allerton 2014 [21].

In this paper, we study the problem of minimizing power consumption for serving real-time applications over fading wireless channels. We consider an analytical model that incorporates the hard per-packet delay bounds and throughput requirements of real-time clients, the fading and unreliable wireless channels, and the optional usage of power control algorithms. We demonstrate that minimizing power consumption can be formulated as a linear programming problem. However, such a problem involves exponentially many constraints, and many parameters are either not available or difficult to compute. Therefore, standard techniques for solving linear programming problems cannot be applied.

We then propose a simple online algorithm that jointly controls various variables involved in serving real-time applications. The algorithm has very low complexity, and it only requires the knowledge of a small number of parameters that are directly available without any computations. We prove that this simple online algorithm satisfies the requirements of all real-time clients. We also prove the difference between the total power consumption of this algorithm and that of an optimum offline algorithm is upper-bounded by a constant, and the constant can be made arbitrarily small. In other words, our algorithm solves the linear programming problem without knowing its parameters.

We also study the system with both real-time and non-real-time clients since in practical networks, these two kinds of clients usually coexist. The throughput requirements of non-real-time clients are assumed to be elastic, and non-real-time clients receive certain utility based on their actual throughputs. We extend our algorithm to achieve low energy consumption and high utility of non-real-time clients. The extended algorithm also offers provable performance guarantees.

The performance of the online algorithm is further evaluated simulations. We have implemented the algorithms in ns-2. Simulation results suggest important insights in choosing various parameters of the online algorithm. They also demonstrate that the proposed algorithm indeed achieves low energy consumption and provides satisfactory services to all real-time clients.

The rest of the paper is organized as follows. Section II summarizes existing related work. Section III introduces our system model and the linear programming formulation. Section IV establishes technical backgrounds for developing the online algorithm. Section V introduces our online algorithm, and Section VI studies the performance of the algorithm. Section VII further extends the online algorithm for non-real-time clients. Section VIII demonstrates our simulation results. Finally, Section IX concludes the paper.

II. Related Work

There have been several studies about real-time wireless networks. Hou, Borkar, and Kumar [4] investigated the problem of scheduling multiple unicast flows with hard deadlines, and some extensions of this work are included in [5] and [6]. Jaramillo and Srikant [7] studied the real time scheduling problem for wireless networks with both real-time and non-real-time clients. Li and Eryilmaz [8] and Mao and Shroff [11] investigated the real time scheduling problem in multihop wireless networks.

Energy optimal control in wireless has been wildly studied. Neely, Modiano, and Rohrs [14] studied the problem for time varying wireless networks and introducing Lyapunov optimization theory. Lin, Lin, and Shroff [9] developed energy efficient algorithm for multihop wireless networks. Yao and Giannakis [20] proposed a low complexity energy efficient scheduling for wireless sensor network and they investigated the tradeoff between energy consumption and delay. L. Venturino et al [19] investigated energy efficiency problem in downlink OFDMA networks with base station coordination. Kemal, Cemil and Ender [3] developed algorithms to maximize energy efficiency in multi-cell systems and they further considered minimum throughput requirement for each client. Nevertheless, none of these works take into consideration of networks with hard deadlines. Miao, Mao and Cassandras [12] studied the energy efficient scheduling in real-time wireless network. However, their policies are inefficient when dealing with long term flows.

Berry and Gallager [1] explored the problem of energy-efficient scheduling with delay constraint, nevertheless, their work only considered the wireless channel with a single user. Stine and Veciana [17] proposed algorithms to improve energy efficiency in centralized wireless networks. Qin et al [15] studied the energy minimizing problem for periodic traffic. However, it requires the knowledge of the relation between energy consumption and transmission rate. Chen, Mitra, and Neely [2] proposed energy efficient scheduling with individual delay constraints. Hassan and Assaad [18] provided a solution within a lower bound and a upper bound for the optimal energy consumption in wireless networks with hard deadlines for each packet. Salodkar et al [16] proposed an online energy efficient algorithm under the framework of constrained Markov decision processes. Nevertheless, none of them provided the computational efficiency in their works.

III. System Model and Problem Formulation

Consider a wireless system with N clients with real-time traffic, numbered as $\{1, 2, ..., N\}$, and one access point (AP). Time is slotted with slot $t \in \{1, 2, ...\}$ such that each slot is the duration of one packet transmission. Time slots are further grouped into intervals where each interval consists of T consecutive time slots in $(kT, (k+1)T]$ with $k \in \{0, 1, ...\}$. We assume that each client generates one packet at the beginning of each interval, and each packet has a hard delay bound of T time slots, i.e., packets generated at the beginning of an interval need to be delivered before the end of the interval. Packets that are not delivered on time are dropped.

We model the unreliable fading wireless channel as an $i.i.d$ process with a finite state space C over intervals. In each interval, the channel stays in state $c \in C$ with probability α_c, and changes its state independently at the beginning of the next interval. Further, we assume that the AP can get instant knowledge on channel states. Let $c(k)$ be the channel state in interval k. We denote the channel reliability of client n under channel state c by $p_{n,c}$, meaning that every transmission for n under c is correctly received with probability $p_{n,c}$. We further consider that the AP may employ some power control algorithm that determines the transmission power used for each client under each channel state. We use $e_{n,c}$ to denote the energy needed to transmit a packet for n under c.

The performance of a client is measured by its *timely-throughput*, defined as the long-term average number of packets delivered for the client per interval. Each client n requires a hard timely-throughput bound of at least q_n packets per interval.

In this paper, we aim to design scheduling algorithm that minimizes the total power consumption while satisfying the timely-throughput requirements of all clients. Let $q_{n,c}$ be the timely-throughput of client n under a particular channel state c. Since channel state c occurs with probability α_c, we have that the overall timely-throughput of n is $\sum_c \alpha_c q_{n,c}$, and hence we require $\sum_c \alpha_c q_{n,c} \geq q_n$. Further, Hou et al. [4] have established the following theorems:

Theorem 1. *Let $w_{n,c}$ be the long-term average number of time slots that the AP transmits for client n under channel state c per interval. The timely-throughput of client n under channel state c is at least $q_{n,c}$ if and only if $w_{n,c} \geq \frac{q_{n,c}}{p_{n,c}}$.*

Theorem 2. *It is feasible to provide a timely-throughput of $q_{n,c}$ to each client n under channel state c if and only if $\sum_{n \in S} \frac{q_{n,c}}{p_{n,c}} \leq T - I_{S,c}$, $I_{S,c}$ is the average number of time slots that are forced to be idle when only the subset S of clients are present. $I_{S,c}$ can be formally written as $\mathbb{E}[(T - \sum_{n \in S} \gamma_{n,c})^+]$, where $\gamma_{n,c}$ is a geometric random variable with mean $1/p_{n,c}$, and $x^+ \triangleq \max\{0, x\}$.*

By Theorem 1, the total power consumption needed to provide a timely-throughput of $q_{n,c}$ for each n and c is $\sum_c \frac{q_{n,c}}{p_{n,c}} e_{n,c} \alpha_c$. Therefore, the problem of minimizing power consumption while satisfying timely-throughput requirements can be written as the following optimization problem:

$$\underset{q_{n,c}}{\text{Minimize}} \quad \sum_{n,c} \frac{q_{n,c}}{p_{n,c}} e_{n,c}\alpha_c \tag{1}$$

$$\text{subject to} \quad \sum_{n\in S} \frac{q_{n,c}}{p_{n,c}} \le T - I_{S,c}, \quad \forall S,\ \forall c, \tag{2}$$

$$\sum_{c} q_{n,c}\alpha_c \ge q_n, \quad \forall n. \tag{3}$$

While this problem is a linear optimization problem, it can be intractable to solve due to the following reasons: First, the channel state distribution α_c may not be available to the clients and AP. Second, there are exponentially many feasibility constraints in (2). Third, it may be difficult to compute each $I_{S,c}$ directly. However, in the following, we demonstrate that there exists a linear time online algorithm that not only satisfies the timely-throughput requirements of all clients, but also achieve a power consumption that is within a constant gap from the optimum.

IV. Preliminaries

Finding an online algorithm for the problem (1) - (3) involves two parts: finding suitable $q_{n,c}$ for all n, c, and finding a scheduling policy that actually provides a timely-throughput of $q_{n,c}$ for each n, c. We call the former problem the *virtual flow control problem* and the latter the *scheduling problem*.

We introduces two variables $q_{n,c}(k)$ and $w_{n,c}(k)$ to capture the AP's solutions to the virtual flow control problem and the scheduling problem, defined as follows. At the beginning of each interval k, the AP observes the current channel state $c(k)$, and it computes the value of $q_{n,c(k)}(k) \in [0,1]$. The value of $q_{n,c(k)}(k)$ can be interpreted as the adjustment of $q_{n,c(k)}$ and the AP tries to find the optimal $q_{n,c(k)}$ for problem (1) - (3). In particular, the AP effectively chooses $q_{n,c} = \liminf_{K\to\infty} \frac{\sum_{k=0}^{K-1} q_{n,c}(k)}{\sum_{k=0}^{K-1} \mathbb{I}_{c(k)=c}}$ as its solutions to the problem (1) - (3), where $q_{n,c}(k) = 0$, if $c \ne c(k)$, and $\mathbb{I}$ is the indicator function, we will prove the optimality of this solution in the following sections.

The AP also determines a scheduling policy of transmitting packets in interval k. We let $w_{n,c(k)}(k)$ be the number of time slots that the AP transmits for client n in interval k. Recall that each transmission for n is successful with probability $p_{n,c}$, and there is only one packet to be delivered for n in each interval. Hence, the value of $w_{n,c(k)}(k)$ is determined by not only the scheduling policy but also random events in interval k. We set $w_{n,c}(k) = 0$ if $c \ne c(k)$. By the definition of $I_{S,c}$, we have that $\mathbb{E}[\sum_{n\in S} w_{n,c(k)}(k)] \le T - I_{S,c(k)}$, for all $S \subseteq \{1, 2, \dots, N\}$, under any scheduling policy.

To capture whether the solutions to the virtual flow control problem and the scheduling problem satisfy the timely-throughput requirements of all clients, we define two *virtual queues* as follows:

$$H_{S,c}(k+1) = H_{S,c}(k) + \sum_{n\in S} \frac{q_{n,c}(k)}{p_{n,c}} - \sum_{n\in S} w_{n,c}(k), \tag{4}$$

and

$$Q_n(k+1) = [Q_n(k) + q_n - q_{n,c(k)}(k)]^+, \tag{5}$$

with $H_{S,c}(0) = 0$, and $Q_n(0) = 0$, for all S, n, and c.

We note the definitions of virtual queues are very similar to those of Lagrange multipliers corresponding to (2) and (3), as used in many online algorithms based on dual decomposition [10] or Lyapunov optimization [13]. However, using the standard definition of Lagrange multipliers, we should have defined $H_{S,c}(k+1) = [H_{S,c}(k) + \sum_{n\in S} \frac{q_{n,c}(k)}{p_{n,c}} - (T - I_{S,c(k)})]^+$, which involves the computation of $I_{S,c}$. On the other hand, our definition of $H_{S,c}$ in (4) is very easy to calculate, as it is solely based on the control decisions of the base station and random events in the interval. Further, we have

Lemma 1. *If* $\limsup_{K\to\infty} \mathbb{E}[|H_{S,c}(K)|]/K = 0$*, and* $\limsup_{K\to\infty} \mathbb{E}[Q_n(K)]/K = 0$*, for all* S*,* c *and* n*, then the timely-throughput of client* n *is at least* q_n*. Further, the vector* $[q_{n,c}] \triangleq [\liminf_{K\to\infty} \frac{\sum_{k=0}^{K-1} q_{n,c}(k)}{\sum_{k=0}^{K-1} \mathbb{I}_{c(k)=c}}]$ *satisfies constraints in (2) and (3), and the total power consumption is at most* $\sum_c \alpha_c e_{n,c} \frac{\bar{q}_{n,c}}{p_{n,c}}$*, where* $[\bar{q}_{n,c}] \triangleq [\limsup_{K\to\infty} \frac{\sum_{k=0}^{K-1} q_{n,c}(k)}{\sum_{k=0}^{K-1} \mathbb{I}_{c(k)=c}}]$.

Proof. Fix $S = \{n\}$, then we have

$$\begin{aligned}
0 &= \limsup_{K\to\infty} \frac{\mathbb{E}[|H_{\{n\},c}(K)|]}{K} \\
&\ge \limsup_{K\to\infty} \frac{1}{K}\Big(\sum_{k=0}^{K-1} \frac{\mathbb{E}[q_{n,c}(k)]}{p_{n,c}} - \sum_{k=0}^{K-1} \mathbb{E}[w_{n,c}(k)]\Big),
\end{aligned}$$

$$\begin{aligned}
\Rightarrow \liminf_{K\to\infty} \frac{\sum_{k=0}^{K-1} \mathbb{E}[w_{n,c}(k)]}{K} &\ge \limsup_{K\to\infty} \sum_{k=0}^{K-1} \frac{\mathbb{E}[q_{n,c}(k)]}{p_{n,c}}/K \\
&\ge \liminf_{K\to\infty} \sum_{k=0}^{K-1} \frac{\mathbb{E}[q_{n,c}(k)]}{p_{n,c}}/K,
\end{aligned}$$

$$\Rightarrow w_{n,c} \ge \frac{q_{n,c}}{p_{n,c}}.$$

Therefore, the timely-throughput of n under c is at least $q_{n,c}$, by Theorem 1. We also have

$$0 = \limsup_{K\to\infty} \frac{\mathbb{E}[Q_n(K)]}{K} \ge q_n - \liminf_{K\to\infty} \frac{1}{K}\mathbb{E}[\sum_{k=0}^{K-1} q_{n,c(k)}(k)]$$

$$\Rightarrow q_n \le \sum_c \alpha_c q_{n,c},$$

and the overall timely-throughput of n is at least q_n.

On the other hand,

$$\begin{aligned}
0 &= \limsup_{K\to\infty} \frac{\mathbb{E}[|H_{\{n\},c}(K)|]}{K} \\
&\ge \limsup_{K\to\infty} \frac{1}{K}\Big(\sum_{k=0}^{K-1} \frac{\mathbb{E}[q_{n,c}(k)]}{p_{n,c}} - \sum_{k=0}^{K-1} \mathbb{E}[w_{n,c}(k)]\Big),
\end{aligned}$$

$$\begin{aligned}
\Rightarrow \liminf_{K\to\infty} \frac{\sum_{k=0}^{K-1} \mathbb{E}[w_{n,c}(k)]}{K} &\ge \limsup_{K\to\infty} \sum_{k=0}^{K-1} \frac{\mathbb{E}[q_{n,c}(k)]}{p_{n,c}}/K \\
&\ge \liminf_{K\to\infty} \sum_{k=0}^{K-1} \frac{\mathbb{E}[q_{n,c}(k)]}{p_{n,c}}/K.
\end{aligned}$$

The total power consumption is then at most $\sum_{n,c} e_{n,c} \limsup_{K\to\infty} \frac{\sum_{k=0}^{K-1}\mathbb{E}[w_{n,c}(k)]}{K} \leq \sum_c \alpha_c e_{n,c}\frac{\bar{q}_{n,c}}{p_{n,c}}$.

Finally, we show that $\sum_{n\in S}\frac{q_{n,c}}{p_{n,c}} \leq T - I_{S,c}$. For any S, c, we have

$$\begin{aligned}
0 &= \limsup_{K\to\infty}\frac{\mathbb{E}[|H_{S,c}(K)|]}{K}\\
&\geq \limsup_{K\to\infty}\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}[\sum_{n\in S}\frac{q_{n,c}(k)}{p_{n,c}} - \sum_{n\in S} w_{n,c}(k)]\\
&\geq \limsup_{K\to\infty}\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}[(\sum_{n\in S}\frac{q_{n,c}(k)}{p_{n,c}} - (T - I_{S,c}))\mathbb{I}_{c(k)=c}]\\
&\Rightarrow \sum_{n\in S}\frac{q_{n,c}}{p_{n,c}} \leq T - I_{S,c}.
\end{aligned}$$

□

We say that $H_{S,c}(k)$ and $Q_n(k)$ are *mean rate stable* if $\limsup_{K\to\infty}\mathbb{E}[|H_{S,c}(K)|]/K = 0$, and $\limsup_{K\to\infty}\mathbb{E}[Q_n(K)]/K = 0$.

V. Online Algorithms

In this section, we develop an online algorithm that jointly solves the virtual flow control problem and the scheduling problem based on Lyapunov theory. Let $\Theta(t) \triangleq [H(k),\ Q(k)]$ be a collective vector of the virtual queues $H(k) = [H_{S,c}(k)]$, and $Q(k) = [Q_n(k)]$. Define a Lyapunov function as:

$$L(\Theta(k)) \triangleq \frac{1}{2}\left(\delta\sum_{S,c} H^2_{S,c}(k) + \sum_n Q^2_n(k)\right), \tag{6}$$

where $\delta \geq 0$ is a predefined parameter and it is used as a weight of $H_{S,c(k)}$ to balance the difference of accumulation rates of $H_{S,c(k)}$ and $Q_n(k)$. We have the following lemma:

Lemma 2. *At each time interval k, for any control decisions applied in the network and constant V, we have the following inequality:*

$$\begin{aligned}
&\mathbb{E}[L(\Theta(k+1)) - L(\Theta(k))|\Theta(k)]\\
&+ V\mathbb{E}[\sum_{n,c}\frac{q_{n,c}(k)}{p_{n,c}}e_{n,c}|\Theta(k),\ c(k)]\\
\leq\ &\mathbb{E}[\delta\sum_{S,c} H_{S,c(k)}(k)(\sum_{n\in S}\frac{q_{n,c}(k)}{p_{n,c}} - \sum_{n\in S} w_{n,c}(k))|\Theta(k),\ c(k)]\\
&+ \mathbb{E}[\sum_n Q_n(k)(q_n - q_{n,c(k)}(k))|\Theta(k),\ c(k)]\\
&+ V\mathbb{E}[\sum_{n,c}\frac{q_{n,c}(k)}{p_{n,c}}e_{n,c}|\Theta(k),\ c(k)] + B,
\end{aligned} \tag{7}$$

where B is a finite constant.

Proof. See Appendix X-A □

The first term in the left hand side (LHS) of (7) is known as the Lyapunov drift. The second term can be explained as follows: if $H_{S,c}(k)$ and $Q_n(k)$ are mean rate stable for all S, n and c, then, as shown in Lemma 1, the long-term average of $\sum_{n,c}\frac{q_{n,c}(k)}{p_{n,c}}e_{n,c}$ is the average power consumption. We therefore refer the LHS of (7) as "drift plus weighted power consumption".

Next, we aim to minimize the right hand side (RHS) of (7). By this minimization, all $H_{S,c}(k)$ and $Q_n(k)$ are mean rate stable, and the average power consumption is within a constant difference from the optimum, see Section IV.

Given $\Theta(k)$ and $c(k)$, the RHS of (7) can be rewritten as

$$\begin{aligned}
&\sum_n \mathbb{E}[\frac{q_{n,c(k)}}{p_{n,c(k)}}(\sum_{S:n\in S}\delta H_{S,c(k)}(k) + Ve_{n,c(k)} - Q_n(k)p_{n,c(k)})]\\
&- \delta\sum_n \mathbb{E}[w_{n,c(k)}\sum_{S:n\in S} H_{S,c(k)}(k)] + \mathbb{E}[\sum_n Q_n(k)q_n] + B.
\end{aligned}$$

With this representation, it is clear that the following policies minimize the RHS of (7):

Virtual Flow Control: At each time interval k, the AP sets $q_{n,c(k)}(k)$ as:

$$q_{n,c(k)}(k) = \mathbb{I}_{\{\sum_{S:n\in S}\delta H_{S,c(k)}(k) + Ve_{n,c(k)} \leq Q_n(k)p_{n,c(k)}\}} \tag{8}$$

Scheduling: At each time interval k, the AP employs a scheduling policy that maximizes

$$\mathbb{E}[\sum_n\sum_{S:n\in S} H_{S,c(k)}(k)w_{n,c(k)}(k)]. \tag{9}$$

To employ such policies, it seems that the AP needs to track the exponentially many variables $[H_{S,k}(k)]$. Instead, we let $D_{n,c}(k) = \sum_{S:n\in S} H_{S,c}(k)$, and we have the following evolution of $D_{n,c}(k)$:

$$\begin{aligned}
D_{n,c}(k+1) =\ &D_{n,c}(k) + \sum_{S:n\in S}(\sum_{m\in S}\frac{q_{m,c}(k)}{p_{m,c}} - \sum_{m\in S} w_{m,c}(k))\\
=\ &D_{n,c}(k) + 2^{N-2}(\frac{q_{n,c}(k)}{p_{n,c}} - w_{n,c}(k))\\
&+ 2^{N-2}(\sum_m\frac{q_{m,c}(k)}{p_{m,c}} - \sum_m w_{m,c}(k)).
\end{aligned} \tag{10}$$

The last equality holds as there are 2^{N-1} subsets that contain n, and 2^{N-2} subsets that contain both n and m, for some $m \neq n$. We can then rewrite (8) and (9) as $q_{n,c(k)}(k) = \mathbb{I}_{\{\delta D_{n,c(k)}(k) + Ve_{n,c(k)} \leq Q_n(k)p_{n,c(k)}\}}$ and $\mathbb{E}[\sum_n D_{n,c(k)}(k)w_{n,c(k)}(k)]$, respectively.

We propose employing the *largest debt first policy*, which was proposed in [4] in a different context, to maximize $\mathbb{E}[\sum_n D_{n,c(k)}(k)w_{n,c(k)}(k)]$. In the largest debt first policy, the AP sorts all clients in descending order of $D_{n,c(k)}$ and schedules transmissions according to this ordering. In other words, a client n is only scheduled by the AP if the packets for all clients m with $D_{m,c(k)} > D_{n,c(k)}$ have been delivered. Clients with negative $D_{n,c(k)}$ are not scheduled by the AP.

Theorem 3. *The largest debt first policy maximizes $\mathbb{E}[\sum_n D_{n,c}(k)w_{n,c}(k)|\Theta(k),\ c(k)]$ in each interval k.*

Proof. Without loss of generality, assume $D_{1,c}(k) \geq D_{2,c}(k) \geq \cdots \geq D_{M,c}(k) \geq 0 \geq \cdots \geq D_{N,c}(k)$. Let γ_n be the random variable denoting the number of transmissions the AP has to make for client n before a successful transmission. Recall that $w_{n,c(k)}(k)$ is the number of transmissions that the AP actually makes for client n during the interval. Assuming that $[\gamma_n]$ is known, maximizing $\mathbb{E}[\sum_n D_{n,c}(k)w_{n,c}(k)]$ reduces to solving the following linear programming problem:

$$\begin{aligned}
\text{Max } & \sum_n D_{n,c(k)}(k)w_{n,c(k)}(k) \\
\text{s.t. } & 0 \leq w_{n,c(k)}(k) \leq \gamma_n, \forall n \\
& \sum_n w_{n,c(k)}(k) \leq T.
\end{aligned}$$

One obvious solution is to allocate the first γ_1 time slots to client 1, the next γ_2 time slots to client 2, etc., until all the T time slots are allocated or all packets for clients 1 through M are delivered. This solution is consistent with the largest debt first policy. □

Our joint algorithm for virtual flow control and scheduling is summarized in Alg. 1. We note that our joint algorithm has the following properties: First, its complexity is $O(N \log N + T)$ per interval. Second, it does not need to know the distributions of channel states, α_c. Third, it does not need to compute the values of $I_{S,c}$.

Algorithm 1 Virtual flow control and scheduling

$Q_n \leftarrow 0, \forall n$
$D_{n,c} \leftarrow 0, \forall n, c$
for each interval **do**
  $c \leftarrow$ current channel
  $q_n \leftarrow \mathbb{I}_{\{\delta D_{n,c} + V e_{n,c} \leq Q_n p_{n,c}\}}, \forall n$
  $w_n \leftarrow 0, \forall n$
  Sort all clients such that $D_{1,c} \geq D_{2,c} \geq \dots$
  $m \leftarrow 0$
  for each time slot in the interval **do**
    if $D_{m,c} > 0$ **then**
      Transmit for client m
      $w_m \leftarrow w_m + 1$
      if the transmission is successful **then**
        $m \leftarrow m + 1$
      end if
    end if
  end for
  $Tot \leftarrow 2^{N-2}(\sum_n \frac{q_n}{p_{n,c}} - \sum_n w_n)$
  $D_{n,c} \leftarrow D_{n,c} + 2^{N-2}(\frac{q_n}{p_{n,c}} - w_n) + Tot, \forall n$
end for

VI. Performance Analysis

In this subsection, we study the performance of Alg. 1.

Theorem 4. *For a controlled parameter $V > 0$, the values of $q_{n,c}(k)$ and $w_{n,c}(k)$ under Alg. 1 satisfy:*

$$\sum_n \frac{\bar{q}_{n,c}}{p_{n,c}} e_{n,c}\alpha_c \leq \sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c + \frac{B}{V}, \tag{11}$$

where $\bar{q}_{n,c} = \limsup_{K\to\infty} \frac{\sum_{k=0}^{K-1} \mathbb{E}[q_{n,c}(k)]}{\sum_{k=0}^{K-1} \mathbb{I}_{\{c(k)=c\}}}$ and $q^*_{n,c}$ is the optimal solution for the problem (1)-(3).

Proof. See Appendix X-B □

Theorem 5. *The virtual queues $[H_{S,c}(k)]$ and $[Q_n(k)]$ under Alg. 1 are mean rate stable.*

Proof. See Appendix X-C □

Finally, since $Q_n(k)$ and $H_{S,c}(k)$ are mean rate stable under Alg. 1, we can establish the following theorem, which is a direct result of Lemma 1 and Theorem 4.

Theorem 6. *Alg. 1 provides a timely-throughput of q_n to each client n, and its total power consumption is at most $\sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c + \frac{B}{V}$.*

VII. Incorporating Non-Real-Time Clients

In this section, we consider the scenario where there are real-time clients and non-real-time ones coexisting in the system. Consider a wireless system with N real-time clients, $\{1, 2, ..., N\}$ and M non-real-time clients $\{N+1, N+2, ..., N+M\}$. Each real-time client has a hard per-packet delay bound of T slots and a hard timely-throughput requirement of q_n. On the other hand, non-real-time clients do not have any hard requirements on delay or throughput, and they generate saturated traffic. They then obtain some utilities based on their actual throughputs. We assume that each non-real time client m obtains a utility of $\sum_c U_m(q_{m,c})\alpha_c$ when its throughput is $q_{m,c}$ under channel state c. We assume that $U_m(\cdot)$ is a strictly increasing, strictly concave, and infinitely differentiable function with $U_m(0) = 0$, for all m. The total utility of all non-real-time clients can then be written as $\sum_{m=N+1}^{N+M} \sum_c U_m(q_{m,c})\alpha_c$, and we aim to minimize $\sum_{n=1}^{N+M} \sum_c \frac{q_{n,c}}{p_{n,c}} e_{n,c}\alpha_c - \gamma \sum_{m=N+1}^{N+M} \sum_c U_m(q_{m,c})\alpha_c$ while satisfying the timely-throughput requirements of real-time clients, where $\gamma \geq 0$ is a variable that tradeoff between total utility in the system and power consumption. Many technical proofs are similar to those in previous sections. Hence, we omit all proofs and only report major results in this section.

This problem can be expressed as the following optimization problem:

$$\text{Minimize } \sum_{n=1}^{N+M} \sum_c \frac{q_{n,c}}{p_{n,c}} e_{n,c}\alpha_c - \gamma \sum_{m=N+1}^{N+M} \sum_c U_m(q_{m,c})\alpha_c \tag{12}$$

$$\text{Subject to} \sum_{n\in S} \frac{q_{n,c}}{p_{n,c}} \leq T - I_{S,c}, \quad \forall S \subseteq \{1, 2, \dots, N\}, \ \forall c \tag{13}$$

$$\sum_{n=1}^{N} \frac{q_{n,c}}{p_{n,c}} + \sum_{m=N+1}^{M} \frac{q_{m,c}}{p_{m,c}} \leq T, \quad \forall c \tag{14}$$

$$\sum_c q_{n,c}\alpha_c \geq q_n, \quad \forall 1 \leq n \leq N \tag{15}$$

As shown in [4], (13) is the condition that it is feasible to provide a timely-throughput of $q_{n,c}$ to client n under channel state c. (14) is the rate constraint on the AP. A similar analysis as that in [4] can then establish the conditions (13) - (15) are necessary and sufficient for feasibility.

We next introduce two new control variables, we let $q_{m,c}(k) \geq 0$ represent the number of admitted packets of each non-real-time client m in interval k, if $c = c(k)$, and $q_{m,c}(k) = 0$ if $c(k) \neq c$. And let $w_{m,c}(k)$ be the number of time slots that the AP transmits packets of non-real time client m, if $c = c(k)$, and $w_{m,c}(k) = 0$ if $c(k) \neq c$. Define:

$$J_{m,c}(k+1) = J_{m,c}(k) + \frac{q_{m,c}(k)}{p_{m,c}} - w_{m,c}(k). \tag{16}$$

The purpose of introducing $J_{m,c}(k)$ is to guarantee that the allocation of resources to non-real-time clients will not exceed the capacity of the system. Specifically, adding (4) with $S = \{1, 2, ..., n\}$ and (16) from $m = 1$ to $m = M$ then we have:

$$\begin{aligned} &H_{\{1,2,\ldots,N\},c}(k+1) + \sum_{m=1}^{M} J_{m,c}(k+1) \\ =&H_{\{1,2,\ldots,N\},c}(k) + \sum_{m=1}^{M} J_{m,c}(k) + \sum_{n=1}^{N+M} \frac{q_{n,c}(k)}{p_{n,c}} - \sum_{n=1}^{N+M} w_{n,c}(k) \end{aligned} \tag{17}$$

Notice that $\sum_{n=1}^{N+M} w_{n,c}(k) \leq T$ for all $k = 0, 1, 2, ...$ If $H_{S,c}(k)$ and $J_{m,c}(k)$ are *mean rate stable*, by Lemma1, the long term average of $q_{n,c}(k)$ and $q_{m,c}(k)$ satisfies condition (14).

We then aggregate the three types of virtual queues together by defining: $\Theta(k) \triangleq [H_{S,c}(k),\ J_{m,c}(k),\ Q_n(k)]$. We employ the Lyapunov function defined as below:

$$L(\Theta(k)) \triangleq \frac{1}{2}\left(\delta \sum_{S,c} H_{S,c}^2(k) + \sum_{m,c} J_{m,c}^2(k) + \sum_{n} Q_n^2(k)\right). \tag{18}$$

Lemma 3. *At each time interval k, for any control decisions applied in the network, we have the following inequality:*

$$\begin{aligned} &\mathbb{E}[L\Theta(k+1)] - \mathbb{E}[L\Theta(k)] \\ &+ V\mathbb{E}[\sum_{n=1}^{N+M}\sum_{c} \frac{q_{n,c}(k)}{p_{n,c}} e_{n,c}|\Phi(k)] \\ &- \gamma V\mathbb{E}[\sum_{m=N+1}^{N+M}\sum_{c} U_m(q_{m,c}(k))|\Phi(k)] \\ \leq\ & B' + \mathbb{E}[\sum_{n=1}^{N}\sum_{c} \delta D_{n,c}(k)(\frac{q_{n,c}(k)}{p_{n,c}} - w_{n,c}(k))|\Phi(k)] \\ &+ \mathbb{E}[\sum_{n=1}^{N} Q_n(k)(q_n - \sum_{c} q_{n,c}(k))|\Phi(k)] \\ &+ \mathbb{E}[\sum_{m=N+1}^{N+M}\sum_{c} J_{m,c}(k)(\frac{q_{m,c}(k)}{p_{m,c}} - w_{m,c}(k))|\Phi(k)] \\ &+ V\mathbb{E}[\sum_{n=1}^{N+M}\sum_{c} \frac{q_{n,c}(k)}{p_{n,c}} e_{n,c}|\Phi(k)] \\ &- \gamma V\mathbb{E}[\sum_{m=N+1}^{N+M} U_m(q_{m,c}(k))|\Phi(k)], \end{aligned} \tag{19}$$

where $B' \geq 0$ is a constant and $\Phi(k) \triangleq [\Theta(k),\ c(k)]$. □

We aim to minimize the RHS of (19) when given $\Phi(k)$, which can be rewritten as

$$\begin{aligned} \text{RHS of (19)} =& \sum_{n=1}^{N}\sum_{c} \mathbb{E}[\frac{q_{n,c}(k)}{p_{n,c}}(\delta D_{n,c}(k) + Ve_{n,c} - Q_n(k)p_{n,c})] \\ &+ \mathbb{E}[\sum_{m=N+1}^{N+M}\sum_{c}((J_{m,c}(k) + Ve_{m,c})\frac{q_{m,c}(k)}{p_{m,c}} - \gamma VU_m(q_{m,c}(k))] \\ &- \delta\mathbb{E}[\sum_{n=1}^{N}\sum_{c} D_{n,c}(k)w_{n,c}(k)] - \mathbb{E}[\sum_{m=N+1}^{N+M}\sum_{c} J_{m,c}(k)w_{m,c}(k)] \\ &+ \mathbb{E}[\sum_{n} Q_n(k)q_n] + B'. \end{aligned}$$

We then propose the following joint control algorithm: at each interval k, the AP observes the current virtual queue state $\Theta(k) \triangleq [H(k),\ J(k),\ Q(k)]$ and channel state $c(k)$, then it gives the following control actions:

Virtual Flow Control: The AP sets $q_{n,c}(k) = 0$, if $c \neq c(k)$, and

$$q_{n,c}(k) = \mathbb{I}_{\{\delta D_{n,c}(k) + Ve_{n,c} \leq Q_n(k)p_{n,c}\}}, \tag{20}$$

for all real-time clients n and c.

Flow Control: The AP admits $q_{m,c(k)}(k)$ packets from non-real-time client m, where $q_{m,c(k)}(k)$ is a random variable with mean value

$$\mathbb{E}[q_{m,c}(k)] = U_m'^{-1}(\frac{J_{m,c}(k) + Ve_{m,c}}{\gamma Vp_{m,c}}), \tag{21}$$

and finite variance. In the above expression, $U_m'^{-1}(\cdot)$ is the inverse of $U_m'(\cdot)$, which is in turn the derivative of $U_m(\cdot)$. The function $U_m'^{-1}(\cdot)$ exists as $U_m(\cdot)$ is strictly concave and infinitely differentiable.

For example, the AP may choose

$$q_{m,c}(k)=\begin{cases}\lceil\psi_{m,c}(k)\rceil, & \text{w.p } \psi_{m,c}(k)-\lfloor\psi_{m,c}(k)\rfloor\\ \lfloor\psi_{m,c}(k)\rfloor, & \text{w.p } \lceil\psi_{m,c}(k)\rceil-\psi_{m,c}(k),\end{cases} \quad (22)$$

where $\psi_{m,c}(k) \triangleq U_m^{\prime -1}(\frac{J_{m,c}(k)+V}{\gamma V p_{m,c}})$, $\lceil x\rceil$ denotes the smallest integer that is larger than x and $\lfloor x\rfloor$ denotes the largest integer that is smaller than x, respectively.

Scheduling Policy: At each interval k, sorting all $D_{n,c(k)}(k)$ and $J_{m,c(k)}(k)$ in decreasing order, transmit packets according to the sorting and when reach the largest $J_{m,c(k)}(k)$, continuously transmit packets of client m until the end of interval k.

The above algorithm is summarized in Alg.2.

Theorem 7. *Alg.2 satisfies the timely-throughput requirements of all clients. Further, the difference between the value of (12) under the above algorithms and that under an optimum algorithm is bounded by* B'/V. □

Our joint control algorithm for system with real-time and non-real-time clients is summarized in Alg.2.

Algorithm 2 (Virtual) flow control and scheduling

$Q_n \leftarrow 0, D_{n,c} \leftarrow 0, J_{m,c} \leftarrow 0, \forall n, m, c$
for each interval **do**
  $c \leftarrow$ current channel
  $q_n \leftarrow \mathbb{I}_{\{\delta D_{n,c}+Ve_{n,c}\le Q_n p_{n,c}\}}, \forall n$
  $q_m \leftarrow U_m^{\prime -1}(\frac{J_{m,c}+Ve_{m,c}}{\gamma V p_{m,c}}), \forall m$
  $w_n \leftarrow 0, w'_m \leftarrow 0, \forall n, m$
  Sort, and remember, all clients such that $D_{1,c} \geq D_{2,c} \geq \dots$
  and $J_{1,c} \geq J_{2,c} \geq \dots$
  $i \leftarrow 1$
  for each time slot in the interval **do**
    if $D_{i,c} > J_{1,c}$ **and** $D_{i,c} > 0$ **then**
      Transmit for real-time client i
      $w_i \leftarrow w_i + 1$
      if the transmission is successful **then**
        $i \leftarrow i + 1$
      end if
    end if
    if $D_{i,c} < J_{1,c}$ **and** $J_{1,c} > 0$ **then**
      Transmit for non-real-time client 1
      $w'_1 \leftarrow w'_1 + 1$
    end if
  end for
  $Tot \leftarrow 2^{N-2}(\sum_n \frac{q_n}{p_{n,c}} - \sum_n w_n)$
  $D_{n,c} \leftarrow D_{n,c} + 2^{N-2}(\frac{q_n}{p_{n,c}} - w_n) + Tot, \forall n$
  $J_{m,c} \leftarrow J_{m,c} + \frac{q_m}{p_{m,c}} - w'_m, \forall m$
end for

VIII. Simulation Results

We have implemented Alg. 1 and Alg. 2 in ns-2 and a trace-based simulation using our wireless sensor network testbed. In this section, we demonstrate our simulation results. For the simulation of system with only real-time clients, we consider a WiFi system where an AP serves 10 real-time clients. We set the length of an interval to be 20ms, which consists of 32 time slots when the AP transmits at 11 Mb/s. Client n requires a timely-throughput of $(70+n)\%$.

We assume that there are 10 channel states and all channel states occur with the same probability. We consider both cases when the AP may, or may not, employ power control algorithms. When power control algorithms are not used, the AP transmits all packets with unit energy, $e_{n,c}=1$. The channel reliability of client n under channel state c is $(30+10((c+n) \mod 7))\%$. On the other hand, when the AP employs power control algorithms, it chooses proper transmission powers such that the channel reliabilities of all clients are $p_{c,n}=80\%$. The transmission power used for client n under channel state c is $((c+n) \mod 4+1)$.

We measure the performance of various policies with two metrics: average total power consumption and average total deficiency. The deficiency of a client n is defined as $[q_n -$ actual timely-throughput of $n]^+$. Total deficiency is then defined as the sum of deficiencies of all clients. All simulation results are the average of five runs.

In the trase-based simulation, we simulate our policy with the data collected by Telosb sensors in lab environment. Our system has four users located in different places in the lab. We use four different transmission powers to simulate the different channel conditions. Each interval consists of seven slots. Each slot is 50ms. For each channel condition, the transmitter broadcasts a packet in every slot. We record the set of packets that each receiver successfully receives for a total length of 500 intervals. In the simulation, when the AP transmits a packet to a client n with a particular power at time t, we check the trace file of client n, and use the t-th broadcast to determine whether n receives the packet. Due to fading, the successful receptions of packets are not i.i.d. across time.

A. The Choice of δ

We first study the choice of the control parameter δ. We chose various values of δ and study the total deficiency. As we are not concerned about power consumption in this simulation, we set $V=0$, and consider the scenario where the AP uses a fixed transmission power. As shown in Fig. 1, setting $\delta=1$ results in poor performance. The total deficiency remains above 2. In fact, it takes more than 300 seconds for the total deficiency to be below 2. The reason that setting $\delta=1$ results in such poor performance is because $D_{n,c}(k)$ and $Q_n(k)$ change at different rates. By definition, $|D_{n,c}(k+1)-D_{n,c}(k)|$ can be on the order of $2^N T$. On the other hand, $|Q_n(k+1)-Q_n(k)|$ is bounded by 1. With this observation, we envision that setting $\delta=\frac{1}{2^N T}$ and making $\delta D_{n,c}(k)$ and $Q_n(k)$ change at similar rates may lead to better performance. As shown in Fig. 1, choosing $\delta=\frac{1}{2^N T}=1/32768$ indeed achieves the best performance, and the total deficiency converges to 0 very

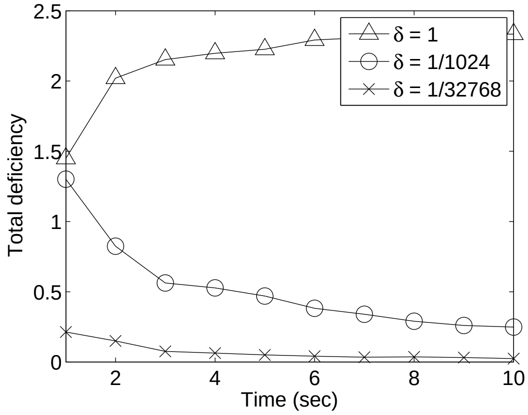


Fig. 1: Total deficiency under various δ.

quickly. Therefore, we set $\delta = 1/32768$ throughout the rest of the section.

B. Performance of System with Real-time Clients

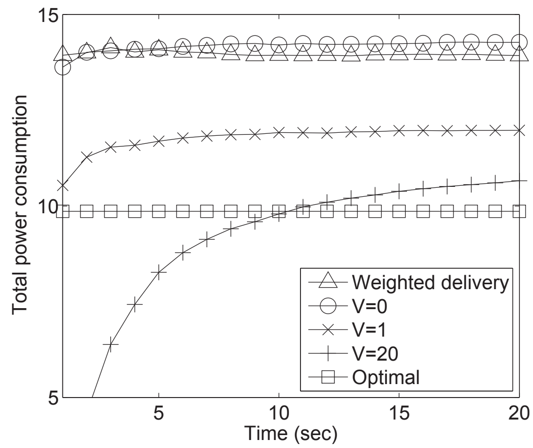
We compare the performance of Alg. 1 for various values of V. We also compare the performance of the weighted delivery policy, which was proposed in [6] and proved to satisfy the requirements of all clients under fading wireless channels as long as they are feasible. In addition, we also use the simplex method in Matlab simulation to show the optimal power consumption.

The ns-2 simulation results when the AP uses a fixed transmission power, and when the AP employs power control algorithms, are presented in Fig. 2 and Fig. 3, respectively. Not surprisingly, the weighted delivery policy and our policy with $V = 0$ have the worst total power consumption, as these two policies do not consider power consumption and only focus on satisfying clients' requirements. Setting $V = 1$ can significantly reduce power consumption, and setting $V = 20$ can further reduce it and make it close to the optimal value. On the other hand, when V is large, the total deficiency converges to zero very slowly. Hence, the choice of V is indeed a tradeoff between power consumption and short-term system performance.

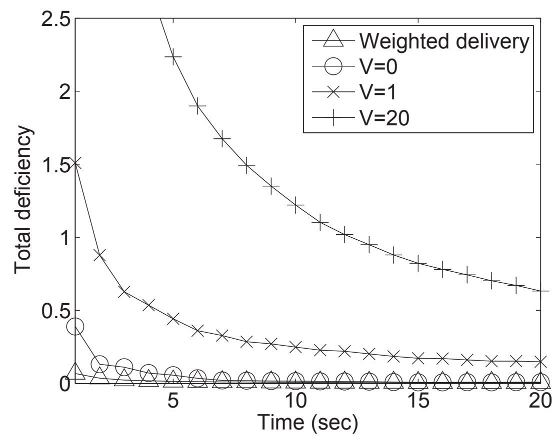
In the trace-based simulation, we employ different transmission powers to simulate different channel conditions. Unlike in the ns-2 simulation, the AP has to estimate the channel reliability for each client. As shown in Fig. 4, our policy can still significantly reduce power consumption by choosing a larger value of V. This implies that Alg. 2 can still perform well in a real world environment.

C. Performance of System including Non-real-time Clients

For the simulation of system with both real-time and non-real-time clients, we consider a WiFi system where an AP serves 7 real-time clients and 5 non-real-time clients. We further consider the scenario where different real-time clients may have different deadline requirements. In this simulation, packets for real-time client 1, 3, 5, 7 have



(a) Total power consumption

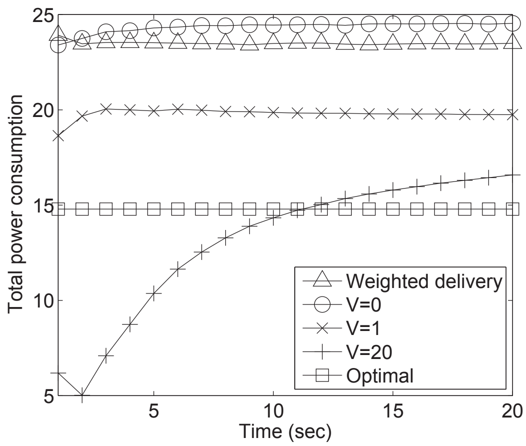


(b) Total deficiency

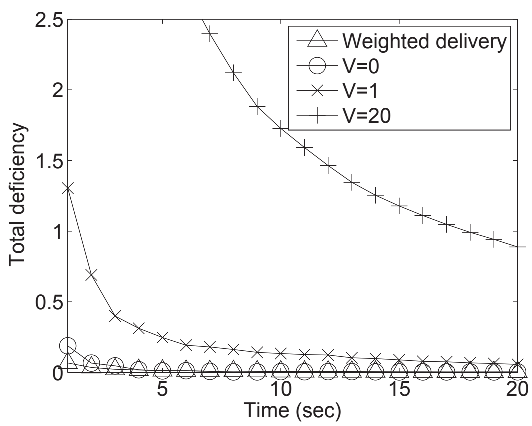
Fig. 2: Simulation results when the AP uses a fixed transmission power.

a deadline of $22.5ms$, which consists of 36 time slots. Packets for real-time client 2, 4, 6 have a deadline of $17.5ms$, which consists of 28 time slots. Moreover, real-time client n requires a timelythroughput of $(70+n)\%$. For non-real-time client m, we choose the utility function as $U_m(q_{m,c}) = log(q_{m,c} + 0.001) + 3$, and we set γ to 10 to represent the weight of the utility.

As in the previous simulation, we assume there are 10 channel states. Real-time clients 1, 3, 5, 7 and non-real-time clients 2, 4 have the same channel state. The channel state remains unchanged within the deadline of real-time clients 1, 3, 5, 7. Similarly, real-time clients 2, 4, 6 and non-real-time clients 1, 3, 5 have the same channel state and it remains unchanged within the deadline of real-time clients 2, 4, 6. When AP doesn't employ power control algorithms, it transmits all packets with unite energy, $e_{n,c} = 1$, where n refers to the index of either real-time clients or non-real-time clients. The channel reliability of client n under channel state c is $(30 + 10((c+n) \mod 7))\%$. When power control algorithms is employed, the AP chooses proper transmission powers such that the

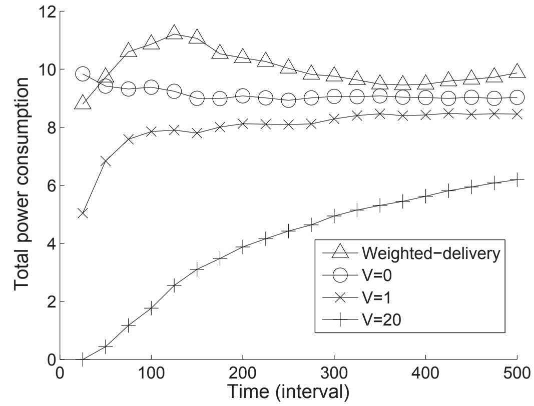


(a) Total power consumption

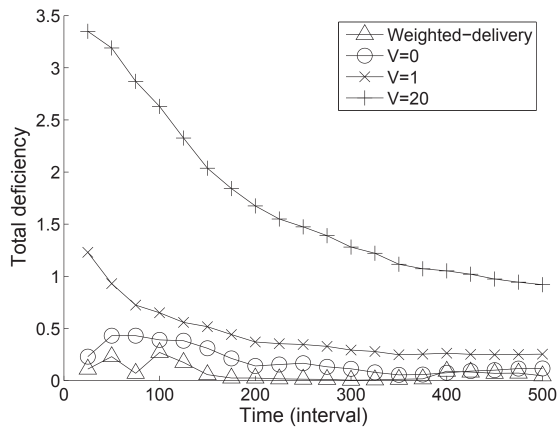


(b) Total deficiency

Fig. 3: Simulation results when the AP employs power control algorithms.



(a) Total power consumption



(b) Total deficiency

Fig. 4: Experimental results.

channel reliabilities of all clients are $p_{c,n} = 80\%$. The transmission power used for client n under channel state c is $((c+n) \mod 4+1)$.

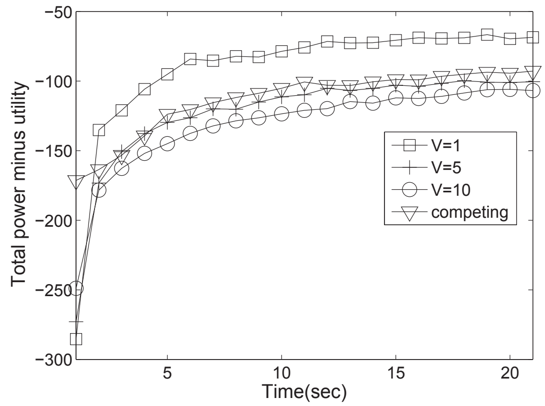
We compare Alg. 2 with various values of V and we also compare it with an competing algorithms and an optimal offline algorithm. The competing algorithm employs Alg. 1 to serve its real-time clients first. If there are remaining slots in the current interval, the AP then serves the non real-time client with the largest $\gamma U_m'(\tilde{q}_{m,c}(k)) - \frac{e_{m,c}}{p_{m,c}}$ until the end of this interval, where $\tilde{q}_{m,c}(k)$ is the average throughput of m before interval k. The intuition of this competing algorithm is that it separates real-time clients and non-real-time clients, and then employs the optimal policy for each of them.

Fig. 5 and Fig. 6 are simulation results when AP uses fixed transmission power and when AP employs power control algorithms, respectively. Once again, we observe that larger V leads to better performance, but at the cost of slower convergence. We also see that the competing algorithm performs worst than our algorithm when V is at least 5. This is because the competing algorithm aims at optimizing the two kinds of clients separately, while our algorithm optimizes them jointly.

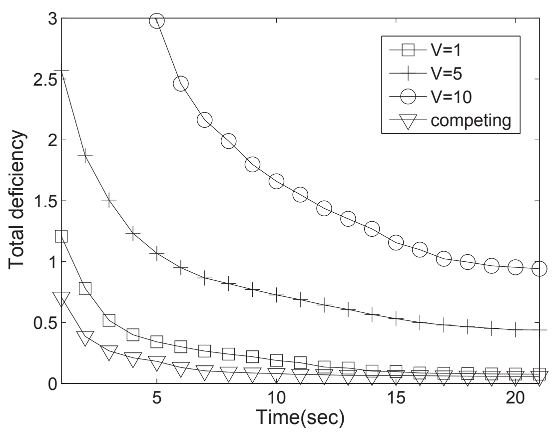
IX. Conclusion

We have studied the problem of providing timely-throughput guarantees to real-time clients with minimum power consumption. While this problem can be formulated as a linear programming problem, such a formulation involves exponentially many constraints, the knowledge of distributions of channel states, and the computation of some complicated parameters. To conquer these challenges, we have proposed a simple algorithm that jointly solves the virtual flow control problem and the scheduling problem. We have proved that this algorithm satisfies the timely-throughput requirement of each client, and its power consumption can be made arbitrarily close to optimum. We further address systems with both real-time and non-real-time clients. We extend our algorithm to provide high utility for each non-real-time client with small total power consumption.

We have implemented and evaluated the performance of our algorithms in ns-2 and in wireless sensor testbed.

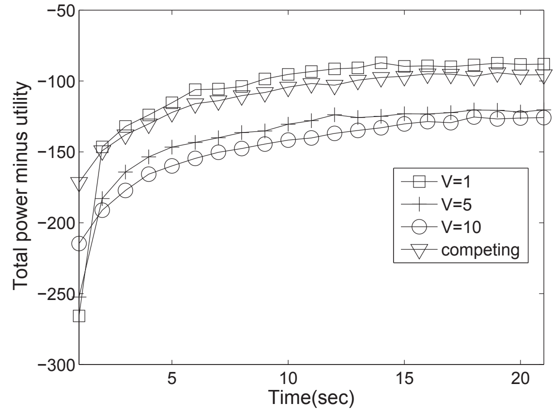


(a) Total power consumption minus utility

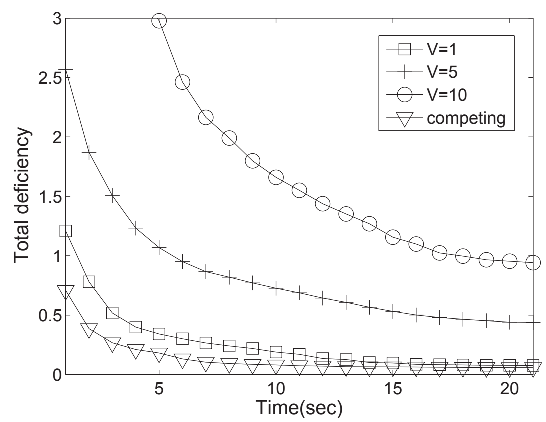


(b) Total deficiency

Fig. 5: When the AP uses a fixed transmission power.



(a) Total power consumption minus utility



(b) Total deficiency

Fig. 6: When the AP employs power control algorithms.

Simulation results provide insights in choosing appropriate parameters for our algorithms to converge quickly. Simulation results also show that our algorithms achieve a substantial reduction on total power consumption and total power consumption minus weighted utility compared to competing policies.

X. Appendix

A. Proof of Lemma 2

Proof. First, we have:

$$\begin{aligned}
& H_{S,c}^2(k+1) - H_{S,c}^2(k) \\
= & \ (H_{S,c}(k) + (\sum_{n\in S} \frac{q_{n,c}(k)}{p_{n,c}} - \sum_{n\in S} w_{n,c(k)}(k)))^2 - H_{S,c}^2(k) \\
= & \ (\sum_{n\in S} \frac{q_{n,c}(k)}{p_{n,c}} - \sum_{n\in S} w_{n,c}(k))^2 \\
& + 2H_{S,c}(k)(\sum_{n\in S} \frac{q_{n,c}(k)}{p_{n,c}} - \sum_{n\in S} w_{n,c}(k)) \\
\leq & \ B_1 + 2H_{S,c}(k)(\sum_{n\in S} \frac{q_{n,c}(k)}{p_{n,c}} - \sum_{n\in S} w_{n,c}(k)), \qquad (23)
\end{aligned}$$

where the last inequality holds because both $\sum_{n\in S} \frac{q_{n,c}(k)}{p_{n,c}}$ and $\sum_{n\in S} w_{n,c}(k)$ are bounded. Similarly, we also have:

$$\begin{aligned}
& Q_n^2(k+1) - Q_n^2(k) \\
= & \ ([Q_n(k) + q_n - q_{n,c(t)}(k)]^+)^2 - Q_n^2(k) \\
\leq & \ (Q_n(k) + q_n - q_{n,c(k)}(k))^2 - Q_n^2(k) \\
= & \ (q_n - q_{n,c(k)}(k))^2 + 2Q_n(k)(q_n - q_{n,c(k)}(k)) \\
\leq & \ B_2 + 2Q_n(k)(q_n - q_{n,c(k)}(k)), \qquad (24)
\end{aligned}$$

where the last inequality holds because $q_{n,c(k)}(k)$ is bounded. Summing (23) and (24) proves the lemma. ☐

B. Proof of Theorem 4

Proof. Since Alg. 1 minimizes the RHS of (7), taking the expected value under Alg. 1 on the LHS of (7) over all $\Theta(k), c(k)$ yields:

$$
\begin{aligned}
&\mathbb{E}[L(\Theta(k+1))] - \mathbb{E}[L(\Theta(k))] + V\mathbb{E}[\sum_{n,c} \frac{q_{n,c}(k)}{p_{n,c}} e_{n,c}] \\
\leq\ & B + \mathbb{E}[\sum_{n,c} D_{n,c}(k)(\frac{q^*_{n,c}\mathbb{I}_{\{c(k)=c\}}}{p_{n,c}} - w_{n,c}(k))] \\
&+ \mathbb{E}[\sum_{n} Q_n(k)(q_n - \sum_{c} q^*_{n,c}\mathbb{I}_{\{c(k)=c\}})] \\
&+ V\mathbb{E}[\sum_{n,c} \frac{q^*_{n,c}\mathbb{I}_{\{c(k)=c\}}}{p_{n,c}} e_{n,c}]
\end{aligned} \tag{25}
$$

Since $q^*_{n,c}$ is the optimal solution for the problem (1)-(3), we have $\mathbb{E}[q_n - \sum_c q^*_{n,c}\mathbb{I}_{\{c(k)=c\}}] \leq 0$, for all n. We also have $\mathbb{E}[\sum_{n,c} \frac{q^*_{n,c}\mathbb{I}_{\{c(k)=c\}}}{p_{n,c}} e_{n,c}] = \sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c$, as channel state c occurs with probability α_c. We can then further simplify the above inequality as

$$
\begin{aligned}
&\mathbb{E}[L(\Theta(k+1))] - \mathbb{E}[L(\Theta(k))] + V\mathbb{E}[\sum_{n,c} \frac{q_{n,c}(k)}{p_{n,c}} e_{n,c}] \\
\leq\ & B + \mathbb{E}[\sum_{n,c} D_{n,c}(k)(\frac{q^*_{n,c}\mathbb{I}_{\{c(k)=c\}}}{p_{n,c}} - w_{n,c}(k))] \\
&+ V\sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c \\
=\ & B + \mathbb{E}[\sum_{n,c} [D_{n,c}(k)]^+(\frac{q^*_{n,c}\mathbb{I}_{\{c(k)=c\}}}{p_{n,c}} - w_{n,c}(k))] \\
&+ V\sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c,
\end{aligned} \tag{26}
$$

where the last equality holds because $w_{n,c}(k) = 0$ if $D_{n,c}(k) < 0$ by Alg. 1.

Since $[q^*_{n,c}]$ satisfies the constraints (2) and (3), there exists a randomized stationary policy that makes scheduling decisions solely based on the channel state and provides a timely-throughput of $q^*_{n,c}$ for each n and c. Let $w^*_{n,c(k)}(k)$ be the number of time slots that the AP transmits for n in interval k, and $w^*_{n,c}(k) = 0$ if $c \neq c(k)$. We then have $\mathbb{E}[\hat{w}_{n,c}(k)] \geq \frac{q^*_{n,c}}{p_{n,c}}\alpha_c$, and:

$$
\begin{aligned}
&\mathbb{E}[L(\Theta(k+1))] - \mathbb{E}[L(\Theta(k))] + V\mathbb{E}[\sum_{n,c} \frac{q_{n,c}(k)}{p_{n,c}} e_{n,c}] \\
\leq\ & B + \sum_{n,c} \mathbb{E}[[D_{n,c}(k)]^+](\frac{q^*_{n,c}}{p_{n,c}}\alpha_c - \mathbb{E}[\hat{w}_{n,c}(k)]) \\
&+ V\sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c \\
\leq\ & B + V\sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c.
\end{aligned} \tag{27}
$$

Summing the above inequality over $k \in \{0, 1, ..., K-1\}$ and dividing by K yields:

$$
\begin{aligned}
&\frac{\mathbb{E}[L(\Theta(K))] - \mathbb{E}[L(\Theta(0))]}{K} + \frac{V}{K}\sum_{k=0}^{K-1}\sum_{n,c} \mathbb{E}[\frac{q_{n,c}(k)}{p_{n,c}} e_{n,c}] \\
&\leq B + V\sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c.
\end{aligned} \tag{28}
$$

Since $L(\Theta(0)) = 0$ and $L(\Theta(k)) \geq 0$, we further have

$$
\begin{aligned}
&\limsup_{K\to\infty} \frac{1}{K}\sum_{k=0}^{K-1}\sum_{n,c} \mathbb{E}[\frac{q_{n,c}(k)}{p_{n,c}} e_{n,c}] \\
=\ & \sum_{n,c} \limsup_{K\to\infty} \frac{\sum_{k=0}^{K-1} \mathbb{E}[q_{n,c}(k)]e_{n,c}}{\sum_{k=0}^{K-1} \mathbb{I}_{\{c(k)=c\}} p_{n,c}} \frac{\sum_{k=0}^{K-1} \mathbb{I}_{\{c(k)=c\}}}{K} \\
=\ & \sum_{n,c} \frac{\bar{q}_{n,c}}{p_{n,c}} e_{n,c}\alpha_c \\
\leq\ & \sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c + \frac{B}{V}.
\end{aligned} \tag{29}
$$

□

C. Proof of Theorem 5

Proof. Since $q_{n,c}(k) \geq 0$ for all n, c, and k, from (27), we get:

$$
\begin{aligned}
&\frac{1}{2}\left(\sum_{S,c} \mathbb{E}[H^2_{S,c}(k)] + \sum_{n} \mathbb{E}[Q^2_n(k)]\right) = \mathbb{E}[L(\Theta(k))] \\
&\leq \mathbb{E}[L(\Theta(0))] + (B + V\sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c)k,
\end{aligned} \tag{30}
$$

and hence

$$
\begin{aligned}
&\frac{1}{2}\mathbb{E}[H^2_{S,c}(k)] \leq \mathbb{E}[L(\Theta(0))] + (B + V\sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c)k \\
&\Rightarrow \mathbb{E}[|H_{S,c}(k)|] \leq \sqrt{2\mathbb{E}[L(\Theta(0))] + 2(B + V\sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c)k}.
\end{aligned}
$$

Dividing by k and let $k \to \infty$ yields:

$$
\begin{aligned}
&\lim_{k\to\infty} \frac{\mathbb{E}[|H_{S,c}(k)|]}{k} \\
&\leq \lim_{k\to\infty} \frac{\sqrt{2\mathbb{E}[L(\Theta(0))] + 2(B + V\sum_{n,c} \frac{q^*_{n,c}}{p_{n,c}} e_{n,c}\alpha_c)k}}{k} = 0.
\end{aligned}
$$

The proof with $Q_n(k)$ is similar, thus, all virtual queues are mean rate stable. □

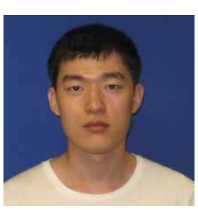
References

[1] Berry, R. A., and Gallager, R. G. Communication over fading channels with delay constraints. *IEEE Trans. on Information Theory 48*, 5 (May 2002), 1135–1149.

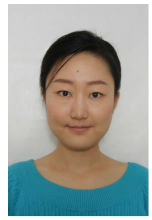
[2] Chen, W., Mitra, U., and Neely, M. Energy-efficient scheduling with individual delay constraints over a fading channel. In *Prof. of WiOpt* (Apr 2007), pp. 1 –10.

[3] Davaslioglu, K., Coskun, C. C., and Ayanoglu, E. New algorithms for maximizing cellular wireless network energy efficiency. *CoRR abs/1610.04950* (2016).

[4] Hou, I.-H., Borkar, V., and Kumar, P. A theory of QoS in wireless. In *Proc. of IEEE INFOCOM* (2009), pp. 486–494.
[5] Hou, I.-H., and Kumar, P. Admission control and scheduling for qos guarantees for variable-bit-rate applications on wireless channels. In *Proc. of MOBIHOC* (2009).
[6] Hou, I.-H., and Kumar, P. Scheduling heterogeneous real-time traffic over fading wireless channels. In *Proc. of IEEE INFOCOM* (2010).
[7] Jaramillo, J., and Srikant, R. Optimal scheduling for fair resource allocation in ad hoc networks with elastic and inelastic traffic. In *Proc. of IEEE INFOCOM* (Mar. 2010), pp. 1 –9.
[8] Li, R., and Eryilmaz, A. Scheduling for end-to-end deadline-constrained traffic with reliability requirements in multi-hop networks. In *Proc. of IEEE INFOCOM* (Apr. 2011), pp. 3065 –3073.
[9] Lin, L., Lin, X., and Shroff, N. B. Low-complexity and distributed energy minimization in multihop wireless networks. *IEEE/ACM Trans. on Networking 18*, 2 (Apr. 2010), 501 – 514.
[10] Lin, X., Shroff, N., and Srikant, R. A tutorial on cross-layer optimization in wireless networks. *Selected Areas in Communications, IEEE Journal on 24*, 8 (2006), 1452–1463.
[11] Mao, Z., Koksal, C. E., and Shroff, N. B. Online packet scheduling with hard deadlines in multihop communication networks. In *Proc. of IEEE INFOCOM* (2013).
[12] Miao, L., Mao, J., and Cassandras, C. G. Optimal energy-efficient downlink transmission scheduling for real-time wireless networks. *IEEE Transactions on Control of Network Systems PP*, 99 (2016), 1–1.
[13] Neely, M. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan and Claypool, 2010.
[14] Neely, M. J., Modiano, E., and Rohrs, C. E. Dynamic power allocation and routing for time varying wireless networks. *IEEE Journal on Selected Areas in Communications 23*, 1 (Jan 2005), 89 –103.
[15] Qin, X., Nijim, M. A. M., Zong, Z., and Bellam, K. Scheduling of periodic packets in energy-aware wireless networks. pp. 210 –217.
[16] Salodkar, N., Karandikar, A., and Borkar, V. A stable online algorithm for energy-efficient multiuser scheduling. *IEEE Trans. on Mobile Computing 9*, 10 (Oct. 2010), 1391 – 1406.
[17] Stine, J. A., and Veciana, G. D. Improving energy efficiency of centrally controlled wireless data networks. *IEEE Trans. on Information Theory 8*, 6 (Nov 2002), 681–700.
[18] Ul Hassan, N.and Assaad, M. Energy efficient causal packet scheduling in wireless fading channels with hard delay constraints. In *Prof. of WCNC* (Apr 2010), pp. 1 –5.
[19] Venturino, L., Zappone, A., Risi, C., and Buzzi, S. Energy-efficient scheduling and power allocation in downlink ofdma networks with base station coordination. *IEEE Transactions on Wireless Communications 14*, 1 (Jan 2015), 1–14.
[20] Yao, Y., and Giannakis, G. B. Energy-efficient scheduling for wireless sensor networks. *IEEE Trans. on Communications 53*, 8 (2005).
[21] Zuo, S., and Hou, I. H. Online scheduling for energy efficiency in real-time wireless networks. In *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)* (Sept 2014), pp. 327–334.



Shuai Zuo is now a PhD student at Texas A&M University, USA.



Han Deng received her M.S. degree in Electrical and Computer Engineering from Oakland University, MI, USA in 2012 and her B.S. in Information Engineering in Beijing Institute of Technology, Beijing, China in 2009. She is now a PhD student at Texas A&M University, USA. Her research interests are in wireless and wired networks and optimization.

I-Hong Hou (S10-M12) received the B.S. in Electrical Engineering from National Taiwan University in 2004, and his M.S. and Ph.D. in Computer Science from University of Illinois, Urbana-Champaign in 2008 and 2011, respectively.

In 2012, he joined the department of Electrical and Computer Engineering at the Texas A&M University, where he is currently an assistant professor. His research interests include wireless networks, wireless sensor networks, real-time systems, distributed systems, and vehicular ad hoc networks.

Dr. Hou received the C.W. Gear Outstanding Graduate Student Award from the University of Illinois at Urbana-Champaign, and the Silver Prize in the Asian Pacific Mathematics Olympiad.