Joint Rate Control and Scheduling for Real-Time Wireless Networks

Shuai Zuo, I-Hong Hou, Tie Liu , Ananthram Swami , and Prithwish Basu

Abstract—**This paper studies wireless networks with multiple real-time flows that have stringent requirements on both per-packet delay and long-term average delivery ratio. Each flow dynamically adjusts its traffic load based on its observation of network status. When the requirements of per-packet delay and delivery ratio are satisfied, each flow obtains some utility based on its traffic load. We aim to design joint rate control and scheduling policies that maximize the total utility in the system. We first show that the problem of maximizing total utility can be formulated as a submodular optimization problem with exponentially many constraints. We then propose two simple distributed policies that require almost no coordination between different entities in the network. The total utilities under these two policies can be made arbitrarily close to the theoretical upper-bound. Extensive simulations also show that they achieve much better performance than state-of-the-art policies.**

I. Introduction

The demands for real-time applications such as video streaming and online gaming in wireless network have been increasing drastically over the decade. According to a recent Cisco report [1], mobile video traffic will increase eleven-fold between 2015 and 2020, and account for three quarters of the overall mobile traffic by 2020. These applications can lead to severe network congestion in wireless networks. Further, real-time applications have some unique features, such as hard per-packet delay requirement and delivery ratio requirement, that separate them from traditional non-real-time applications. Directly employing standard congestion control and resource allocation policies without explicitly addressing these features can thus lead to poor network performance.

In this paper, we study wireless systems that consist of multiple real-time flows where each flow can dynamically adjust its traffic load. Each flow has a hard per-packet delay requirement to meet real-time constraint, as well as a packet delivery ratio requirement to ensure data integrity. When both delay and delivery ratio requirements are satisfied, each flow obtains some utility based on its traffic load. We can consider the application of video call, such as Skype and Facetime, as a motivating example for our problem. In video call, the application can change its traffic load by changing its video encoding scheme, where better video quality leads to larger traffic load. Packets of video call need to be delivered within a certain delay bound, typically around 20ms, or end users may experience severe conversation interruptions due to delay. A high ratio of packets, typically around 90% - 95%, need to be successfully delivered to ensure video frames can be decoded and played smoothly.

We aim to design joint rate control and scheduling policies that achieve the maximum system-wide total utility. We propose an analytical model that incorporates the aforementioned features of real-time applications and the stochastic nature of wireless transmissions. The problem of maximizing total utility can then be modeled as a submodular optimization problem with exponentially many constraints. While there exists polynomial-time algorithms that solve the submodular optimization problem, they still incur high computational complexity and rely on centralized algorithms.

We then propose two distributed policies where traffic load is determined by each flow, packet scheduling is determined by the access point (AP), and there is almost no coordination between flows and the AP. These two policies have very low complexity and can be easily implemented. We prove that both distributed policies satisfy all real-time application requirements. Moreover, both policies achieve a total utility that can be made arbitrarily close to the theoretical upper-bound.

Our proposed policies are further evaluated by ns-2 simulations. We compare our policies against three other policies based on state-of-the-art mechanisms. Simulation results demonstrate that our policies achieve much better performance than other policies under a wide range of different scenarios.

The rest of the paper is organized as follows: Section II summarizes some existing studies. Section III introduces our analytical model. Section IV shows that the problem can be modeled as a submodular optimization problem, and introduces a centralized algorithm to solve it. Section V describes our two proposed distributed policies. Section VI analyzes the performance of our policies and proves that they achieve near-optimal total utility. Section VII demonstrates our simulation results. Finally, Section VIII concludes the paper.

This material is based upon work supported by, or in part by, the U. S. Army Research Laboratory and the U. S. Army Research Office under contract/grant number W911NF-15-1-0279.

Shuai Zuo, I-Hong Hou and Tie Liu are with the Department of ECE, Texas A&M University. Emails: {zuosh029, ihou, tieliu}@tamu.edu

Ananthram Swami is with Army Research Labs. Email: ananthram.swami.civ@mail.mil

Prithwish Basu is with Raytheon BBN Technologies. Email: pbasu@bbn.com

II. Related Work

Since the pioneering work of Kelly [12], the study of controlling networks using an optimization approach has continued for over two decades. Lin and Shroff [14] proposed a joint rate control and scheduling algorithm for non real-time applications in multi-hop wireless networks. Similarly, Eryilmaz and Srikant [4] designed and analyzed a joint algorithm with rate control, routing and scheduling for non real-time clients in wireless networks. Further, Luna et al. [15] studied the utility maximization problem and proposed a joint control algorithm for video streaming applications. However, these works do not consider the deadline requirements in real-time applications.

Within this decade, there have been a number of works aimed at the control of wireless applications with stringent deadline requirements. Hou [6] and Hou et al. [7] proposed feasible scheduling algorithms for unreliable wireless networks subject to deadline constraints. Li and Eryilmaz [13], Xiong et al. [18] and Mao [16] explored the joint control problem for multi-hop wireless networks where clients have deadline constraints. However, these studies assumed that the transmission rates are fixed, whereas the transmission rate for real-time applications may change in response to the network state and in turn affect control decisions.

Chen et al. [3] studied the single user wireless system with real-time video transmission and formulated this problem as a Markov Decision Problem (MDP). However, such MDP approach suffers from the curse of dimensionality and is thus infeasible for systems with multiple users. Kang et al. [11] also focused on a single user system and their work does't provide an analysis of optimality. Huang et al. [10] and Zhao and Lin [20] studied video streaming wireless networks with multiple deadline constraints users. However, [10] didn't provide the optimality guarantees for their heuristic algorithm. The optimality of algorithms in [20] is based on the assumption that the system includes a large number of channels and users.

III. System Model and Problem Formulation

We extend the model in [7] to incorporate dynamic rate control policies for real-time applications. Consider a wireless system with one access point (AP) serving M wireless clients. Each client is associated with a downlink real-time flow that can adaptively adjust its data rate based on network congestion. Time is slotted and indexed as $t = 1, 2, \ldots$, where the duration of a time slot is chosen to be the amount of time needed to transmit a data packet and an ACK. Wireless transmissions can be unreliable, and we say that each transmission for client m is successful with probability p_m. By requiring ACKs for all transmissions, the AP has the feedback information about whether a transmission is successful, and may retransmit the same packet if previous transmissions fail. In the downlink scenario, sources of real-time flows have reliable and wired connections to the AP, and can therefore obtain feedback information, such as channel state and whether a transmission is successful from the AP.

Each real-time flow has some stringent per-packet delay bound. Specifically, we say that the flow of client m generates packets periodically with period T_m, and the delay bound of m is the same as its period. Therefore, if a period of m starts at time t, then all packets of m generated at time t need to be delivered on or before the end of the period at time $t + T_m - 1$. Packets that are not delivered within their respective delay bounds are dropped from the system.

Our theoretical analysis focuses on the special case where all clients have the same $T_m \equiv T$, and all their first periods start at time $t = 1$. We can then group time slots into intervals, where each interval consists of all time slots in $(kT, (k+1)T]$, for some integer k. All flows generate packets at the beginning of each interval, and all these packets are either delivered before the end of the interval, or dropped from the system at the end of the interval. In Section VII, we will demonstrate that our proposed policies achieve good performance even when different clients have different periods.

We consider real-time applications that can adjust their data rates, such as video applications that can choose among different resolutions. At the beginning of each interval, the flow of client m generates r_m packets, where r_m lies in $[0, R_m]$. The quality of experience of client m is then determined by r_m, and we say it receives utility $U_m(r_m)$, where $U_m(\cdot)$ is assumed to be an increasing and strictly concave function. On the other hand, client m requires that at least a portion q_m of its packets are delivered on time. When its flow generates r_m packets per interval, client m requires that its throughput be at least $r_m q_m$ packets per interval. We note that, in our model, delivering more than q_m of client m's packets does not further improve its utility. This assumption is based on existing studies on Skype video quality [2], [19], where they show that the video quality has a significant drop when packet loss rate exceeds $6 \sim 10\%$, but stays rather stable as long as the packet loss rate is below 6%.

We aim to find policies that jointly control the data rate of each flow and the packet scheduling decisions of the AP so as to maximize the total utility, $\sum_m U_m(r_m)$, while ensuring that the throughput of each client m is at least $r_m q_m$. In order to simplify the formulation of this problem, we first divide each client into R_m sub-clients, where each sub-client can generate at most one packet in each interval. We set $p_n = p_m$ and $q_n = q_m$ if n is a sub-client of m. We say that the i-th sub-client of m generates a packet if and only if $r_m \geq i$. Further, when a sub-client n who is the i-th sub-client of m generates a packet, it receives a utility $u_i := U_m(i) - U_m(i-1)$. Let $\mathbb{N}$ be the set of all sub-clients, we then have $\sum_m U_m(r_m) = \sum_{n \in \mathbb{N}} u_n 1(\text{sub-client } n \text{ generates a packet})$, where $1(\cdot)$ is the indicator function. Our problem is to find the optimal subset $\mathbb{O}$ of sub-clients that maximizes $\sum_{n \in \mathbb{O}} u_n$, while ensuring that the throughput of each sub-client $n \in \mathbb{O}$ is at least q_n.

Hou et al. [7] have derived a necessary and sufficient condition for checking whether it is feasible to ensure a throughput of q_n for each $n \in \mathbb{O}$.

Theorem 1: A throughput of q_n for each $n \in \mathbb{O}$ is feasible if and only if

$$\sum_{n \in S} \frac{q_n}{p_n} \leq F(S), \forall S \subseteq \mathbb{O}, \tag{1}$$

where $F(S) := \mathbb{E}[\min\{T, \sum_{n \in S} \gamma_n\}]$, and γ_n is a geometric random variable with mean $\frac{1}{p_n}$. □

By definition, $F(S)$ is increasing in the sense that $F(S_1) \geq F(S_2)$ if $S_2 \subseteq S_1$. The condition (1) is then equivalent to

$$\sum_{n \in S} \frac{q_n}{p_n} 1(n \text{ generates a packet}) \leq F(S), \forall S \subseteq \mathbb{N}.$$

Let $x_n := \frac{q_n}{p_n} 1(n$ generates a packet$)$. The problem of maximizing total utility can then be formulated as

$$\begin{aligned} \text{Max } & \sum_n \frac{u_n p_n}{q_n} x_n \\ \text{s.t. } & \sum_{n \in S} x_n \leq F(S), \ \forall S \\ & x_n \in \{0, \frac{q_n}{p_n}\}, \ \forall n \end{aligned}$$

We relax the last constraint to transform this integer programming problem into a linear programming problem:

$$\text{Max } \sum_n \frac{u_n p_n}{q_n} x_n \tag{2}$$

$$\text{s.t. } \sum_{n \in S} x_n \leq F(S), \ \forall S \tag{3}$$

$$0 \leq x_n \leq \frac{q_n}{p_n}, \ \forall n \tag{4}$$

In this paper, we aim to design joint rate control and scheduling policies that solve problem (2)-(4).

IV. A Centralized Greedy Algorithm

In this section, we propose a centralized greedy algorithm that solves problem (2)-(4) optimally using techniques developed for submodular optimization.

Let Ω be a nonempty finite set and $\mathbb{R}$ be the set of real numbers. A real-valued function $f : 2^{\Omega} \rightarrow \mathbb{R}$ is called *submodular* over the subsets of Ω if, for $\forall X \subset Y \subseteq \Omega$, and $\forall n \notin Y$, we have

$$f(X \cup \{n\}) - f(X) \geq f(Y \cup \{n\}) - f(Y)$$

It has been shown that $F(S) := \mathbb{E}[\min\{T, \sum_{n \in S} \gamma_n\}]$ is a submodular function.

Lemma 2 (Theorem 1, [8]): $F(S)$ is a submodular function over all subsets of $\mathbb{N}$.

Since $F(S)$ is a submodular function, the problem (2)-(4) is called a *submodular optimization problem with vector constraints* [5], and there exists a greedy algorithm that solves the problem. The greedy algorithm works as follows: it first sorts all sub-clients in the order of $\frac{u_1 p_1}{q_1} \geq \frac{u_2 p_2}{q_2} \geq \ldots$. It calculates $x_1, x_2, \ldots x_N$ in this order one by one. The assignment of x_n is according to $x_n = \hat{F}(\{1, 2, \ldots, n\}) - \hat{F}(\{1, 2, \ldots, n-1\})$, where $\hat{F}(\cdot)$ is defined as

$$\hat{F}(S) = \min_{Z \subseteq S} \{F(Z) + \sum_{n \in S \setminus Z} \frac{q_n}{p_n}\}.$$

The greedy algorithm is summarized in Alg.1. It has been shown that the greedy algorithm finds the optimal solution under some mild conditions.

Algorithm 1 Greedy algorithm

```
1: $x_n \leftarrow 0, \forall n$
2: Sort all sub-clients such that $\frac{u_1 p_1}{q_1} \geq \frac{u_2 p_2}{q_2} \geq \ldots$
3: $n \leftarrow 1$
4: $x_n = \hat{F}(\{1, 2, \ldots, n\}) - \hat{F}(\{1, 2, \ldots, n-1\})$
5: $n \leftarrow n + 1$, go to step $4$
```

Theorem 3 (Theorem 3.3, [5]): Let $x_1^*, x_2^*, \ldots$ be the optimal solution to (2)-(4). If $\sum_{n \in \mathbb{N}} x_n^* = F(\mathbb{N})$, then the greedy algorithm produces the optimal solution.

According to [9], given $F(Z)$ for every subset $Z \subseteq S$, $\hat{F}(S)$ can be computed in $O((|\mathbb{N}|^5 EO + |\mathbb{N}|^6) \log |\mathbb{N}|)$ time, where EO is the time to evaluate function $F(Z)$. However, calculating $F(Z)$ is difficult since it does not have a simple closed-form expression. Moreover, the greedy algorithm is a centralized algorithm and it does not provide any solution to packet scheduling. In most real-time applications, rate control occurs at the client side while the scheduling of the transmission of packets naturally occurs at the AP side. Therefore, distributed solutions that only requires limited coordination between the AP and clients are highly desirable.

V. Distributed Policies for Joint Rate Control and Scheduling

In this section, we propose two joint rate control and scheduling policies. One is a fully distributed policy that requires almost no information exchange between the AP and clients. The other policy is similar to the first one, but employs a global variable to achieve a better convergence rate.

Both of our policies have two components: First, each client m determines r_m, which is the number of packets it generates in each interval. This is equivalent to determining the values of x_n in problem (2)-(4). We call this component the *rate control problem*. Second, the AP determines its scheduling decisions so as to provide a throughput of $q_m r_m$ to each client m. We call the second component the *scheduling problem*.

We introduce variable $x_n(k)$ to capture the client's solution to the *rate control problem*, defined as follows: At the beginning of each interval k, The client sets the value $x_n(k) \in \{0, \frac{q_n}{p_n}\}$ for its corresponding sub-client n. If $x_n(k) = \frac{q_n}{p_n}$, sub-client n generates a "real" packet in the k-th interval. If $x_n(k) = 0$, sub-client n generates a "dummy" packet. Therefore, the flow of client m generates a total number of $(\frac{p_m}{q_m} \sum_{n\text{:sub-clients of } m} x_n(k))$ real packets, and $(R_m - \frac{p_m}{q_m} \sum_{n\text{:sub-clients of } m} x_n(k))$ dummy packets. We require m to generate dummy packets so that it always has R_m packets for transmissions in each interval. Define $\bar{x}_n \triangleq \liminf_{K\to\infty} \frac{\sum_{k=1}^{K} \mathbb{E}[x_n(k)]}{K}$ as the long-term average of $x_n(k)$. The long-term average number of real packets generated by sub-client n is then $\frac{p_n}{q_n}\bar{x}_n$, and it requires a throughput of $p_n\bar{x}_n$.

The AP employs a scheduling policy to determine the transmission of packets in each interval. Let $w_n(k)$ be the number of time slots that the AP used for transmitting the packet for sub-client n in interval k. Due to unreliable wireless transmissions, the value of $w_n(k)$ depends on both the employed scheduling policy and the random packet loss events in the interval. Define $\bar{w}_n \triangleq \liminf_{K\to\infty} \frac{\sum_{k=1}^{K} \mathbb{E}[w_n(k)]}{K}$. Hou et al. [7] have established the following relations between $\bar{w}_n$ and throughput of n.

Lemma 4 (Lemma 1, [7]): The throughput of sub-client n is at least $p_n\bar{x}_n$ if and only if $\bar{w}_n \geq \bar{x_n}$.

Given $\{x_n(k)\}$ and $\{w_n(k)\}$, we define a *deficiency* for each sub-client n as follows:

$$D_n(k+1) = D_n(k) + x_n(k) - w_n(k), \tag{5}$$

with $D_n(0) = 0$, for all n. By Lemma 4, the throughput of n is at least $p_n\bar{x}_n$ if $\limsup_{k\to\infty} \frac{D_n(k)}{k} \leq 0$.

A. Fully Distributed Policy

Our first rate control and scheduling policy, named as the *fully distributed policy*, employs the deficiency given in (5) and it works as follows.

Rate Control Policy of Clients: Let $V > 0$ be a control parameter. At the beginning of each interval k, the AP sets $x_n(k)$ to $\frac{q_n}{p_n}$, and generates a real packet for sub-client n, if $u_n \geq \frac{q_n D_n(k)}{p_n V}$. It sets $x_n(k)$ to 0 and generates a dummy packet if $u_n < \frac{q_n D_n(k)}{p_n V}$.

Scheduling Policy of the AP: At the beginning of each interval, the AP sorts all sub-clients according to their deficiencies such that $D_1(k) \geq D_2(k) \geq \dots$ and schedules packet transmissions for the sub-clients according to this order. In particular, in each time slot in the interval, the AP schedules the sub-client n with the largest $D_n(k)$ among those whose packets have not been delivered and $D_n(k) > 0$. If the AP has delivered all packets for all sub-clients with $D_n(k) > 0$, then the AP idles for the remaining time slots in the interval. [1]

Many real-time applications generate packets that have dependencies. For example, in scalable video coding, a video frame is separated into a base layer and several enhancement layers. Enhancement layers cannot be decoded without the base layer, while the base layer can be decoded independently. It is then required to transmit the base layer before transmitting enhancement layers. For these applications, when the AP schedules a sub-client n of client m, it transmits the first undelivered packet of m.

Algorithm 2 Fully distributed policy

```
$D_n \leftarrow 0, \forall n$
for each interval do
    $x_n \leftarrow \frac{q_n}{p_n} 1\{u_n \geq \frac{q_n D_n}{p_n V}\}, \forall n$
    $w_n \leftarrow 0, \forall n$
    Sort all sub-clients such that $D_1 \geq D_2 \geq \dots$
    $j \leftarrow 1$
    for each time slot in the interval do
        if $D_j > 0$ then
            Transmit for $j$
            $w_j \leftarrow w_j + 1$
            if the transmission is successful then
                $j \leftarrow j + 1$
            end if
        end if
    end for
    $D_n \leftarrow D_n + x_n - w_n, \forall n$
end for
```

Our fully distributed policy is summarized in Alg.2. In each interval, the AP computes $D_n(k)$ and $x_n(k)$ for all sub-clients. Consequently, each flow generates packets according to it's corresponding x_ns. After that, the AP only needs to sort packet transmissions for all sub-clients according to their deficiencies. The complexity on the AP's side is $O(|\mathbb{N}| \log |\mathbb{N}|)$. In the next section, we will demonstrate that our policy converges to a point that can be made arbitrarily close to the optimum solution to (2)-(4).

B. Policy with A Small Overhead

We propose a second policy, which is called the policy with a small overhead. In this policy, the AP computes $\Phi(k) \triangleq \sum_n D_n(k)$, and broadcasts its value to all flows. Each flow then determines its rate based on both $\Phi(k)$ and $D_n(k)$. The policy is described as the following:

Rate Control Policy of Clients: At the beginning of each interval k, the AP sets $x_n(k)$ to $\frac{q_n}{p_n}$, and generates a real packet, if $u_n - \frac{q_n \Phi(k)}{p_n V} \geq \frac{q_n D_n(k)}{p_n V}$.

Scheduling Policy of the AP: The scheduling policy is exactly the same as the fully distributed policy.

We note that the policy with a small overhead is very similar to the fully distributed policy. The only difference is that the rate control policy in the policy with a small overhead involves both $\Phi(k)$ and $D_n(k)$. Intuitively, the value of $\Phi(k)$ reflects the congestion of the whole network. By taking $\Phi(k)$ into account, the policy with a small overhead should achieve better convergence rate,

[1] We note that our scheduling policy is not work-conserving in the sense that it does not transmit any packets with $D_n(k) \leq 0$, even when there are no packets with $D_n(k) > 0$ to transmit. This artificial constraint is needed for the performance analysis in Section VI. One can obviously modify the policy by allowing it to schedule packets with $D_n(k) \leq 0$. Whether such a modified policy remains optimal, and whether it improves convergence speed, is an interesting question for future research.

which we indeed demonstrate by simulations in Section VII. The detailed algorithm for the policy with a small overhead is described in Alg. 3 and it has a complexity of $O(|\mathbb{N}|\log|\mathbb{N}|)$.

Algorithm 3 Policy with a small overhead

```
$D_n \leftarrow 0, \forall n$
for each interval do
    $\Phi \leftarrow \sum_n D_n$
    $x_n \leftarrow \frac{q_n}{p_n} 1\{u_n - \frac{q_n \Phi}{p_n V} \geq \frac{q_n D_n}{p_n V}\}, \forall n$
    $w_n \leftarrow 0, \forall n$
    Sort all sub-clients such that $D_1 \geq D_2 \geq \dots$
    $j \leftarrow 1$
    for each time slot in the interval do
        if $D_j > 0$ then
            Transmit for $j$
            $w_j \leftarrow w_j + 1$
            if the transmission is successful then
                $j \leftarrow j + 1$
            end if
        end if
    end for
    $D_n \leftarrow D_n + x_n - w_n, \forall n$
end for
```

VI. Performance Analysis

In this section, we show that our two proposed policies are both asymptotically optimal and feasible, as formally stated in Theorem 5 and Theorem 6, respectively.

Theorem 5: Let $\bar{x}_n$ be the value of $\liminf_{K\to\infty} \frac{\sum_{k=1}^{K} \mathbb{E}[x_n(k)]}{K}$ under the fully distributed policy. For any controlled parameter $V > 0$, the long-term average total utility achieved under this policy satisfies:

$$\sum_n \frac{u_n p_n}{q_n} \bar{x}_n \geq \sum_n \frac{u_n p_n}{q_n} x_n^* - \frac{B}{V},$$

where $\{x_n^*\}$ is the optimal solution to problem (2)-(4), and B is a bounded constant. Further, the fully distributed policy satisfies all throughput requirements by providing a throughput of at least $p_n \bar{x}_n$ to each sub-client n.

Proof: Let $D(k)$ be the vector containing all $D_n(k)$, and define a Lyapunov function with respect to $\{D_n(k)\}$ as follows:

$$L(D(k)) \triangleq \frac{1}{2}\left(\sum_n D_n^2(k)\right).$$

We then have, under any policy and constant V,

$$\begin{aligned}
&\mathbb{E}[L(D(k+1)) - L(D(k))|D(k)] \\
=\ & \frac{1}{2}\sum_n \mathbb{E}[(D_n(k) + (x_n(k) - w_n(k)))^2 - D_n^2(k)|D(k)] \\
=\ & \frac{1}{2}\sum_n \mathbb{E}[(x_n(k) - w_n(k))^2 + 2D_n(k)(x_n(k) - w_n(k))|D(k)] \\
\leq\ & B + \mathbb{E}[\sum_n D_n(k)(x_n(k) - w_n(k))|D(k)],
\end{aligned}$$

where B is a bounded constant since $x_n(k)$ and $w_n(k)$ are both bounded for all n. Therefore,

$$\begin{aligned}
&\mathbb{E}[L(D(k+1)) - L(D(k)) - V\sum_n \frac{u_n p_n}{q_n} x_n(k)|D(k)] \\
\leq\ & \mathbb{E}[\sum_n D_n(k)(x_n(k) - w_n(k)) - V\sum_n \frac{u_n p_n}{q_n} x_n(k)|D(k)] + B \\
=\ & \mathbb{E}[\sum_n (D_n(k) - V\frac{u_n p_n}{q_n}) x_n(k)|D(k)] \\
& - \mathbb{E}[\sum_n D_n(k) w_n(k)|D(k)] + B \\
\leq\ & \mathbb{E}[\sum_n (D_n(k)^+ - V\frac{u_n p_n}{q_n}) x_n(k)|D(k)] \\
& - \mathbb{E}[\sum_n D_n(k)^+ w_n(k)|D(k)] + B,
\end{aligned}$$

where $D_n(k)^+ := \max\{0, D_n(k)\}$, and the last inequality follows because the fully distributed policy does not transmit for sub-clients with negative $D_n(k)$.

Recall that $0 \leq x_n(k) \leq \frac{q_n}{p_n}$, and the fully distributed policy sets $x_n(k) = \frac{q_n}{p_n}$ if and only if $D_n(k) - V\frac{u_n p_n}{q_n} \leq 0$. Therefore, the fully distributed policy minimizes $\mathbb{E}[\sum_n (D_n(k)^+ - V\frac{u_n p_n}{q_n}) x_n(k)|D(k)]$. Moreover, Theorem 3 in [21] has shown that the scheduling policy employed by the fully distributed policy maximizes $\mathbb{E}[\sum_n D_n(k)^+ w_n(k)|D(k)]$. Thus, the fully distributed policy minimizes $\mathbb{E}[\sum_n (D_n(k)^+ - V\frac{u_n p_n}{q_n}) x_n(k)|D(k)] - \mathbb{E}[\sum_n D_n(k)^+ w_n(k)|D(k)]$.

Next, consider a policy that sets $x_n(k) = x_n^*$, and employs a stationary randomized scheduling policy that provides a throughput of $p_n x_n^*$. Such a scheduling policy must exist since $\{x_n^*\}$ satisfies all constraints in problem (2)-(4). Let w_n^* be the expected number of transmissions that the policy schedules n in an interval. By Lemma 4, $w_n^* \geq x_n^*$.

We now have, under the fully distributed policy,

$$\begin{aligned}
&\mathbb{E}[L(D(k+1)) - L(D(k)) - V\sum_n \frac{u_n p_n}{q_n} x_n(k)|D(k)] \\
\leq\ & \mathbb{E}[\sum_n (D_n(k)^+ - V\frac{u_n p_n}{q_n}) x_n(k)|D(k)] \\
& - \mathbb{E}[\sum_n D_n(k)^+ w_n(k)|D(k)] + B, \\
\leq\ & \sum_n (D_n(k)^+ - V\frac{u_n p_n}{q_n}) x_n^* - \sum_n D_n(k)^+ w_n^* + B \\
\leq\ & B - \sum_n V\frac{u_n p_n}{q_n} x_n^*.
\end{aligned}$$

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

$$\begin{aligned}
&\frac{\mathbb{E}[L(D(K))] - \mathbb{E}[L(D(0))]}{K} - \frac{V}{K}\sum_{k=0}^{K-1}\sum_n \mathbb{E}[\frac{u_n p_n}{q_n} x_n(k)] \\
\leq & B - V\sum_n \frac{u_n p_n}{q_n} x_n^*. \qquad (6)
\end{aligned}$$

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

$$\liminf_{K\to\infty} \frac{1}{K}\sum_{k=0}^{K-1}\sum_n \mathbb{E}[\frac{u_n p_n}{q_n} x_n(k)]$$
$$=\sum_n \frac{u_n p_n}{q_n}\bar{x}_n \geq \sum_n \frac{u_n p_n}{q_n} x_n^* - \frac{B}{V}.$$

Next, we show that the fully distributed policy provides a throughput of at least $p_n\bar{x}_n$ to each sub-client n.

Since $0 \leq x_n(k) \leq \frac{q_n}{p_n}$ for all n and k, from (6), we get

$$\begin{aligned}&\frac{1}{2}\sum_n \mathbb{E}[D_n^2(K)] = \mathbb{E}[L(D(K))]\\ \leq&(B + \frac{V}{K}\sum_{k=0}^{K-1}\sum_n \mathbb{E}[\frac{u_n p_n}{q_n} x_n(k)])K\\ \leq&(B + V\sum_n u_n)K,\end{aligned}$$

and hence

$$\mathbb{E}[|D_n(K)|] \leq \sqrt{2(B + V\sum_n u_n)K}.$$

Dividing by K and letting $K \to \infty$ yields

$$\lim_{K\to\infty}\frac{\mathbb{E}[|D_n(K)|]}{K} \leq \lim_{K\to\infty}\frac{\sqrt{2(B+V\sum_n u_n)K}}{K} = 0,$$

and hence the throughput of each sub-client n is at least $p_n\bar{x}_n$. ■

Theorem 6: Let $\bar{x}_n$ be the value of $\liminf_{K\to\infty}\frac{\sum_{k=1}^{K}\mathbb{E}[x_n(k)]}{K}$ under the policy with a small overhead. For any control parameter $V > 0$,

$$\sum_n \frac{u_n p_n}{q_n}\bar{x}_n \geq \sum_n \frac{u_n p_n}{q_n} x_n^* - \frac{B'}{V},$$

where B' is a bounded constant. Further, the fully distributed policy satisfies all throughput requirements.

Proof: The proof consists of two parts: we first prove the optimality of the policy, and then prove that it provides a throughput of at least $p_n\bar{x}_n$ to each sub-client n.

For each subset $S \subseteq \mathbb{N}$ of sub-clients, define $J_S(k) \triangleq \sum_{n\in S} D_n(k)$. $J_S(k)$ can be expressed recursively as follows,

$$J_S(k+1) = J_S(k) + \sum_{n\in S} x_n(k) - \sum_{n\in S} w_n(k),$$

with $J_S(0) = 0$. Let $D(k)$ be the vector containing all $D_n(k)$. Define a Lyapunov function as:

$$L(D(k)) \triangleq \frac{1}{2^{|\mathbb{N}|-1}}\sum_S J_S^2(k).$$

We then have,

$$\begin{aligned}&\mathbb{E}[L(D(k+1)) - L(D(k))|D(k)]\\ \leq\;& \mathbb{E}[\sum_S \frac{1}{2^{|\mathbb{N}|-2}} J_S(k)(\sum_{n\in S} x_n(k) - \sum_{n\in S} w_n(k))|D(k)] + B'\\ =\;& \mathbb{E}[\frac{1}{2^{|\mathbb{N}|-2}}\sum_n (x_n(k) - w_n(k))(\sum_{S:n\in S} J_S(k))|D(k)] + B'\\ =\;& \mathbb{E}[\frac{1}{2^{|\mathbb{N}|-2}}\sum_n (x_n(k) - w_n(k))(\sum_{S:n\in S}\sum_{l\in S} D_l(k))|D(k)] + B',\end{aligned} \tag{7}$$

where B' is a bounded constant. For each n, there exists $2^{|\mathbb{N}|-1}$ subsets of $\mathbb{N}$ that contains n. Also, for any two different sub-clients n and l, there exists $2^{|\mathbb{N}|-2}$ subsets of $\mathbb{N}$ that contains both n and l. Therefore, $\sum_{S:n\in S}\sum_{l\in S} D_l(k) = 2^{|\mathbb{N}|-2}(D_n(k) + \Phi(k))$, where $\Phi(k) = \sum_l D_l(k)$ as defined by the policy with a small overhead. Therefore,

$$\begin{aligned}&\mathbb{E}[L(D(k+1)) - L(D(k)) - V\sum_n \frac{u_n p_n}{q_n} x_n(k)|D(k)]\\ \leq\;& \mathbb{E}[\sum_n (\Phi(k) + D_n(k) - V\frac{u_n p_n}{q_n}) x_n(k)|D(k)]\\ &- \mathbb{E}[\sum_n (\Phi(k) + D_n(k)) w_n(k)|D(k)] + B'\\ \leq\;& \mathbb{E}[\sum_n ([\Phi(k) + D_n(k)]^+ - V\frac{u_n p_n}{q_n}) x_n(k)|D(k)]\\ &- \mathbb{E}[\sum_n (\Phi(k) + D_n(k))^+ w_n(k)|D(k)] + B'.\end{aligned}$$

By its design, the policy with a small overhead minimizes

$$\begin{aligned}&\mathbb{E}[\sum_n ([\Phi(k) + D_n(k)]^+ - V\frac{u_n p_n}{q_n}) x_n(k)|D(k)]\\ &- \mathbb{E}[\sum_n (\Phi(k) + D_n(k))^+ w_n(k)|D(k)].\end{aligned}$$

Now, consider a policy that sets $x_n(k) = x_n^*$ for all k, and employs a stationary randomized scheduling policy so that the average number of transmissions scheduled for sub-client n in each interval is $w_n^* \geq x_n^*$. Therefore, for any given $D(k)$, we have, under the policy with a small overhead,

$$\begin{aligned}&\mathbb{E}[L(D(k+1)) - L(D(k)) - V\sum_n \frac{u_n p_n}{q_n} x_n(k)|D(k)]\\ \leq\;& \sum_n ([\Phi(k) + D_n(k)]^+ - V\frac{u_n p_n}{q_n}) x_n^*\\ &- \sum_n (\Phi(k) + D_n(k))^+ w_n^* + B'\\ \leq\;& B' - V\sum_n \frac{u_n p_n}{q_n} x_n^*,\end{aligned}$$

and therefore,

$$\begin{aligned}&\frac{\mathbb{E}[L(D(K))] - \mathbb{E}[L(D(0))]}{K} - \frac{V}{K}\sum_{k=0}^{K-1}\sum_n \mathbb{E}[\frac{u_n p_n}{q_n} x_n(k)]\\ \leq& B' - V\sum_n \frac{u_n p_n}{q_n} x_n^*.\end{aligned} \tag{8}$$

Since $L(D(0)) = 0$ and $L(D(k)) \geq 0$, we thus have

$$\liminf_{K\to\infty}\frac{1}{K}\sum_{k=0}^{K-1}\sum_{n}\mathbb{E}[\frac{u_n p_n}{q_n}x_n(k)] = \sum_{n}\frac{u_n p_n}{q_n}\bar{x}_n \geq \sum_{n}\frac{u_n p_n}{q_n}x_n^*$$

On the other hand, notice that $0 \leq x_n(K) \leq \frac{q_n}{p_n}$ for all n and K. Therefore, from (8), we get

$$\begin{aligned}&\frac{1}{2^{N-1}}\sum_{S}\mathbb{E}[J_S^2(K)] = \mathbb{E}[L(D(K))]\\ \leq&(B' + \frac{V}{K}\sum_{k=0}^{K-1}\sum_{n}\mathbb{E}[\frac{u_n p_n}{q_n}x_n(k)])K\\ \leq&(B' + V\sum_{n}u_n)K,\end{aligned}$$

and hence

$$\lim_{K\to\infty}\frac{\mathbb{E}[|J_S(K)|]}{K} \leq \lim_{K\to\infty}\frac{\sqrt{2^{N-1}(B' + V\sum_{n}u_n)K}}{K} = 0,$$

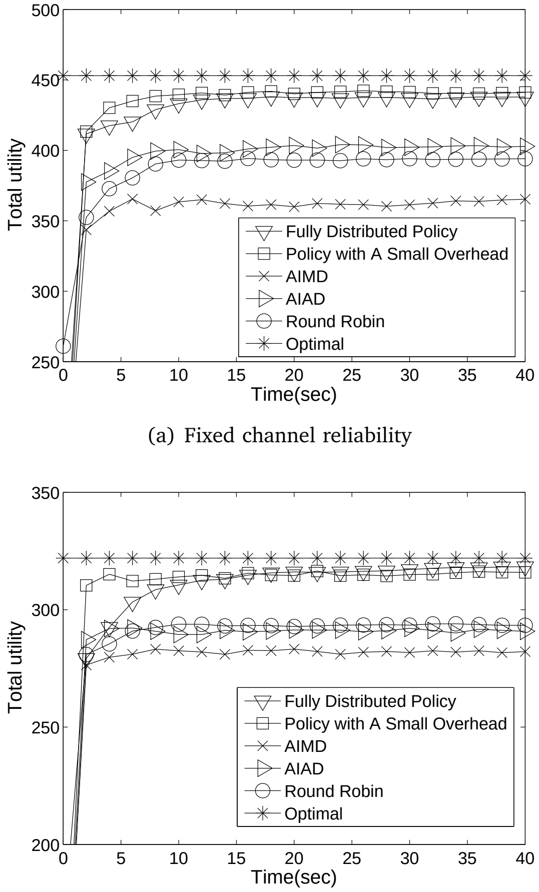
for any subset S. This also implies that $\lim_{K\to\infty}\frac{\mathbb{E}[|D_n(K)|]}{K} = 0$, and the throughput of sub-client n is at least $p_n\bar{x}_n$. ■

VII. Simulation Results

We have implemented the fully distributed policy and the policy with a small overhead in ns-2, and we demonstrate our simulation results in this section. We consider both cases where all clients have the same period, $T_m \equiv T, \forall m$, and where different clients have different periods.

We compare our policies with three competing policies. The first policy applies an Additive Increase Multiplicative Decrease (AIMD) rate control scheme. In particular, client m increases r_m by 1 if all its real packets in the previous interval are delivered on time, and sets r_m to $\lceil r_m/2 \rceil$, otherwise. The second policy applies an Additive Increase Additive Decrease (AIAD) rate control scheme, where client m increases r_m by 1 if all its real packets in the previous interval are delivered on time, and decreases r_m by 1, otherwise. Both policies employ the scheduling policy of our fully distributed policy for packet scheduling. The third policy employs the rate control policy of the fully distributed policy, but uses round-robin for packet scheduling.

We evaluate these policies through three different metrics: The first is the total utility, $\sum_m U_m(r_m)$, where r_m is the average number of packets that client m generates per interval. The second is the total deficiency, defined as $\frac{\sum_n D_n(k)^+}{k}$. The AP delivers the required throughput, $q_m r_m$, to each client m if and only if $\frac{\sum_n D_n(k)^+}{k}$ converges to 0. Finally, we also evaluate the total variance of the number of packets that client m generates in each interval. Many existing studies have shown that a high variance in data rate can result in poor quality of experience (QoE). A desirable policy should therefore achieve low total variance.



(a) Fixed channel reliability

(b) Fixed QoS requirements

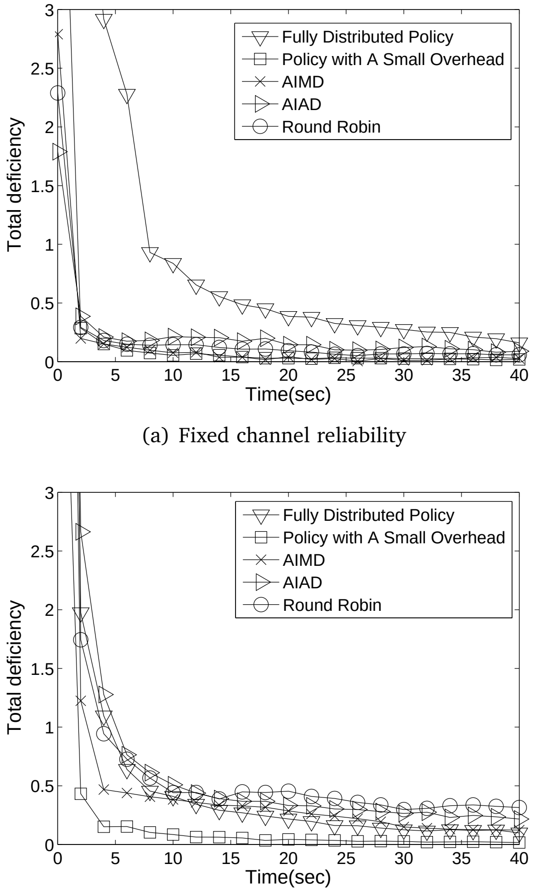
Fig. 1: Total utility

A. Homogeneous Period

We consider a WiFi system where the AP serves 4 clients, each of which has $R_m = 6$. Therefore, there are 24 sub-clients in the system. The interval length is set to be $20ms$, and each interval is composed of 25 consecutive time slots when the AP transmits at 11 Mb/s.

We consider two different scenarios. In the first scenario, clients have different service requirements, but the same channel reliability. According to the study in [17], we choose the utility function of client m as $5\log(r_m + 0.001) + (15 + m)r_m$. In addition, we set $q_m = (83 + 2m)\%$ and $p_m = 0.65$. In the second scenario, clients have different wireless channel conditions, but the same service requirements. We set $p_1 = 0.65$, $p_2 = 0.55$, $p_3 = 0.6$ and $p_4 = 0.5$. The utility function of m is $5(\log(r_m + 0.001) + 15$, and we choose $q_m = 0.9$ for all clients.

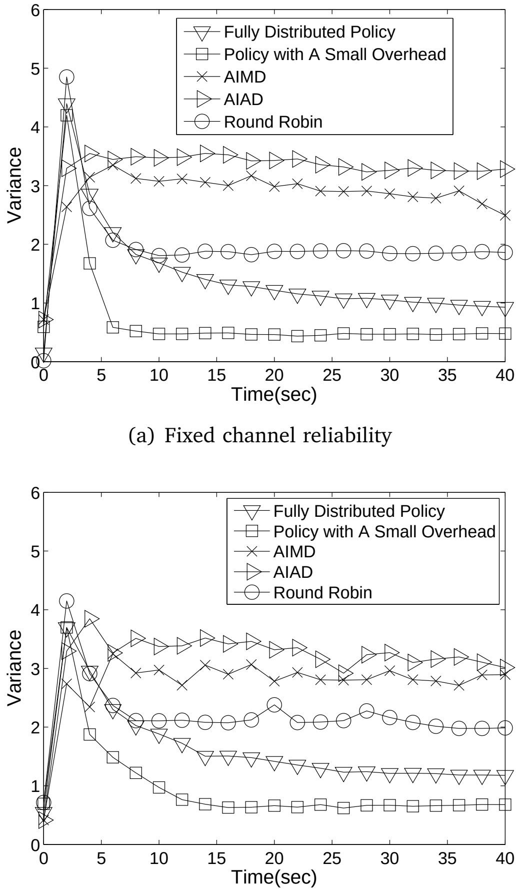
Simulation results on the three performance metrics are shown in Figs. 1, 2 and 3. In both scenarios, our policies achieve higher total utility than the other three policies, and also satisfy the throughput requirements of all flows. Further, our policies result in performance with smaller variances than other policies. It is also observed in the simulation that the policy with a small overhead seems to have better transitional performance than the



(a) Fixed channel reliability

(b) Fixed QoS requirements

Fig. 2: Total deficiency



(a) Fixed channel reliability

(b) Fixed QoS requirements

Fig. 3: Variance of r_1

fully distributed policy in that its total utility converges to the optimal value faster, and its total deficiency also converges to 0 faster.

B. Heterogeneous Periods

Next, we consider systems where different flows have different periods T_m. We consider a WiFi system where the AP serves 4 clients. Clients 1 and 2 have $R_m = 3$ and $T_m = 20ms$, while clients 3 and 4 have $R_m = 7$ and $T_m = 25ms$. The utility functions are $U_1(r_1) := 5\log(r_1 + 0.001) + 15r_1$, $U_2(r_2) := 5\log(r_2 + 0.001) + 17r_2$, $U_3(r_3) := 5\log(r_3 + 0.001) + 23r_3$ and $U_4(r_4) := 5\log(r_4 + 0.001) + 25r_4$. We set $q_m = (83 + 2m)\%$ and $p_m = 0.65$.

The simulation results are shown in Figs.4, 5, and 6. Both of our policies still perform better than the other three policies in both total utility and total variance. This result suggests that our proposed policies can still achieve better performance when clients have different periods.

VIII. Conclusion

We investigated the utility maximization problem for real-time applications with stringent deadlines. We formulate this problem as a submodular optimization problem.

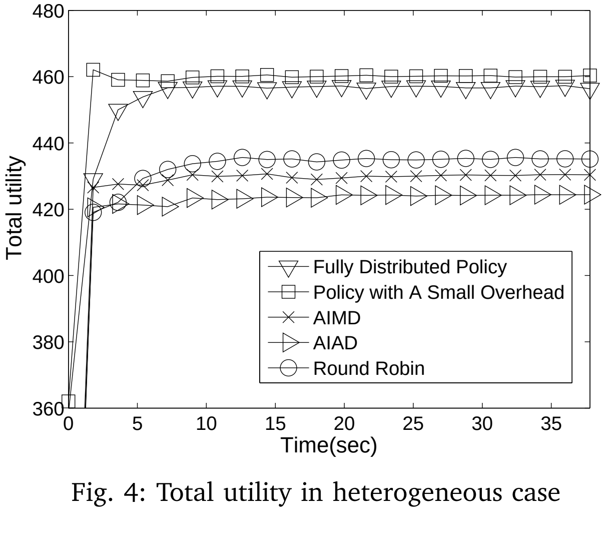


Fig. 4: Total utility in heterogeneous case

Since there is no standard technique that can solve it efficiently, we propose two asymptotically optimal policies, both of which have very low complexity. The performance of our policies is further evaluated via ns-2 simulation. All simulation results show that our policies achieve better performance than three other competing policies.

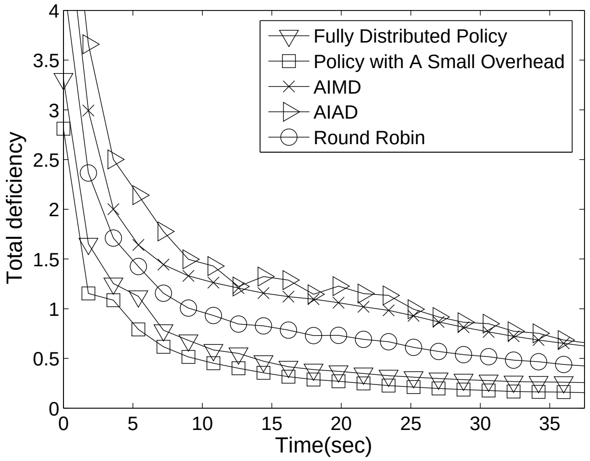


Fig. 5: Total deficiency in heterogeneous case

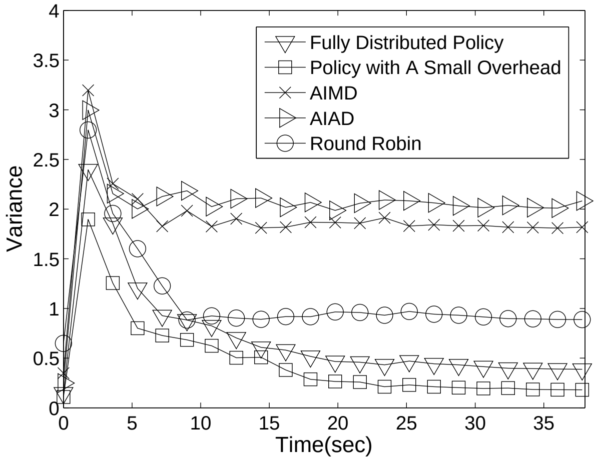


Fig. 6: Variance in heterogeneous case

References

[1] Cisco visual networking index: Global mobile data traffic forecast update, 2015c2020 white paper. In *Cisco* (Feb 2016).

[2] Asiri, A., and Sun, L. Performance analysis of video calls using skype. *Advances in Communications, Computing, Networks and Security Volume 10* (2013), 155.

[3] Chen, C., Heath, R. W., Bovik, A. C., and de Veciana, G. Adaptive policies for real-time video transmission: A markov decision process framework. In *IEEE international Conference on Image Processing* (Sep 2011).

[4] Eryilmaz, A., and Srikant, R. Joint congestion control, routing and mac for stability and fairness in wireless networks. *IEEE Journal on Selected Areas in Communications 24* (2006), 1514–1524.

[5] Fujishige, S. Submodular functions and optimization. *Annals of Discrete Mathematics 23*, 1 (Jan 2005), 89 –103.

[6] Hou, I.-H. Scheduling heterogeneous real-time traffic over fading wireless channels. *IEEE/ACM Trans. on Networking 22*, 5 (Oct. 2014), 1631 – 1644.

[7] Hou, I.-H., Borkar, V., and Kumar, P. A theory of QoS for wireless. In *Proc. of IEEE INFOCOM* (2009), pp. 486–494.

[8] I-H. Hou, A. Truong, S. C., and Kumar, P. Optimality of periodwise static priority policies in real-time communications. In *Proc. of CDC (invited)* (Orlando, FL, Dec 2011).

[9] Iwata, S., and Orlin, J. A simple combinatorial algorithm for submodular function minimization. In *Proc. of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms* (2009).

[10] Ji, X., Huang, J., Chiang, M., Lafruit, G., and Catthoor, F. Scheduling and resource allocation for svc streaming over ofdm downlink systems. *IEEE Trans. Circuits Syst. Video Techn 19*, 10 (2009), 1549 –1555.

[11] Kang, S. H., and Zakhor, A. Packet scheduling algorithm for wireless video streaming. In *Packet Video 2002* (Apr 2002).

[12] Kelly, F. Charging and rate control for elastic traffic, 1997.

[13] 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.

[14] Lin, X., and Shroff, N. Joint rate control and scheduling in multihop wireless networks. In *in Proceedings of IEEE Conference on Decision and Control* (Dec. 2004).

[15] Luna, C. E., Kondi, L. P., and Katsaggelos, A. K. Maximizing user utility in video streaming applications. *IEEE Trans. Circuits Syst. Video Techn. 13*, 2 (2003), 141–148.

[16] 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).

[17] Schroeder, D., Essaili, A. E., Steinbach, E., Staehle, D., and Shehada, M. Low-complexity no-reference psnr estimation for h.264/avc encoded video. In *Packet Video Workshop (PV), 2013 20th International* (Dec 2013).

[18] Xiong, H., Li, R., and Eryilmaz, A. Delay-aware cross-layer design for network utility maximization in multi-hop networks. *IEEE Journal on Selected Areas in Communications 29*, 5 (May 2011), 951 –959.

[19] Zhang, X., Xu, Y., Hu, H., Liu, Y., Guo, Z., and Wang, Y. Profiling skype video calls: Rate control and video quality. In *INFOCOM, 2012 Proceedings IEEE* (2012), IEEE, pp. 621–629.

[20] Zhao, S., and Lin, X. Rate-control and multi-channel scheduling for wireless live streaming with stringent deadlines. In *Proc. of IEEE INFOCOM* (2014).

[21] Zuo, S., and Hou, I.-H. Online scheduling for energy efficiency in real-time wireless networks. In *Proc. of Allerton* (2014).

Shuai Zuo is currently pursuing the Ph.D. degree with Texas A&M University, College Station, TX, USA. His research interests include wireless networking, real-time systems 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.

Tie Liu (S99–M06–SM15) was born in Jilin, China in 1976. He received his B.S. (1998) and M.S. (2000) degrees, both in Electrical Engineering, from Tsinghua University, Beijing, China and a second M.S. degree in Mathematics (2004) and a Ph.D. degree in Electrical and Computer Engineering (2006) from the University of Illinois at Urbana-Champaign. Since August 2006 he has been with Texas A&M University, where he is currently an Associate Professor in the Department of Electrical and Computer Engineering. His primary research interest is in the area of information theory and statistical information processing.

Dr. Liu received an M. E. Van Valkenburg Graduate Research Award (2006) from the University of Illinois at Urbana-Champaign and a CAREER Award (2009) from the National Science Foundation. He was a Technical Program Committee Co-Chair for the 2008 IEEE GLOBECOM, a General Co-Chair for the 2011 IEEE North American School of Information Theory, and an Associate Editor for Shannon Theory for the IEEE Transactions on Information Theory during 2014-2016.

Ananthram Swami is with the US Army Research Laboratory as the Army's Senior Research Scientist (ST) for Network Science. Prior to joining ARL, he held positions with Unocal Corporation, USC, CS-3 and Malgudi Systems. He was a Statistical Consultant to the California Lottery, developed a Matlab-based toolbox for non-Gaussian signal processing. He has held visiting faculty positions at INP, Toulouse., and currently at Imperial College. He received the B.Tech. degree from IIT-Bombay; the M.S. degree from Rice University, and the Ph.D. degree from the University of Southern California (USC), all in Electrical Engineering. Swami's work is in the broad area of network science, with emphasis on wireless communication networks. He is an ARL Fellow and a Fellow of the IEEE.

Prithwish Basu (Sr. Member '09) received a B.Tech. degree in Computer Science and Engineering from Indian Institute of Technology, Delhi in 1996, and his M.S. and Ph.D. in Computer Engineering from Boston University in 1999 and 2003, respectively.

He is currently a Lead Scientist at Raytheon BBN Technologies in the Networking and Communications Technologies business unit. His research interests include network science, wireless ad hoc and sensor networks, and mobile computing. He has co-authored over 100 papers in network-related conferences and journals and has received best paper awards at IEEE NetSciComm and PAKDD conferences. He is an Associate Editor of the IEEE Transactions of Mobile Computing.

Dr. Basu received the MIT Technology Review's TR35 award in 2006 (given to 35 innovators under the age of 35), and was ranked 7th in the Indian National Mathematics Olympiad in 1991.