Queueing Systems with Hard Delay Constraints: A Framework for Real-Time Communication over Unreliable Wireless Channels*†

I-Hong Hou‡ and P. R. Kumar§

May 3, 2011

Abstract

We provide an account of recent work that formulates and addresses problems that arise when employing wireless networks to serve clients that generate real-time flows. From a queueing systems perspective, these problems can be described as single-server problems where there are several customer classes. Customer's balk when their delay exceeds a threshold. There are a range of issues that are of interest. One of the first such issues is to determine what throughput rate vectors are feasible, and to determine the server's schedule. Another is to maximize a utility function of the departure rates of the customer classes.

Real-time flows have a delay bound for each of their packets. It is particularly challenging to provide delay guarantees for real-time flows in wireless networks since wireless transmissions are unreliable. We propose a model that jointly considers the delay bounds of packets, the unreliable wireless channels, and the throughput requirements of clients. We then determine the necessary and sufficient condition for feasibility of the client requirements. The analysis and condition are interesting since this problem gives rise to some new features concerning unavoidable idle times in a system. We further derive an efficient, nearly linear time algorithm for admission control, which precisely determines whether it is feasible to fulfill the requirements of all clients in the system. We also propose two on-line scheduling policies and prove that they can fulfill the requirements of all clients whenever that is feasible.

We next turn to the scenario where the throughput requirements of clients are elastic, but with hard delay bounds. We formulate this as a utility maximization problem, where client utilities are based on their throughputs. We decompose this problem into two subproblems, and

*This material is based upon work partially supported by USARO under Contract Nos. W911NF-08-1-0238 and W-911-NF- 0710287, NSF under Contracts CNS-1035378, CNS-0905397, CNS-1035340, and CCF-0939370, and AFOSR under Contract FA9550-09-0121. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the above agencies.

†Part of this work has been presented at the 2009 IEEE INFOCOM and 2010 IEEE INFOCOM.

‡CSL and Department of CS, University of Illinois, 201 North Goodwin Avenue, Urbana, IL 61801-2302, USA. Email: ihou2@illinois.edu.

§CSL and Department of ECE, University of Illinois, 1308 West Main St., Urbana, IL 61801-2307, USA. Email: prkumar@illinois.edu.

show that this decomposition can be naturally implemented as a bidding game among all clients and the access point, which plays the role of a centralized scheduler. In the bidding game, the strategy of each client is to carry out a simple selfish optimization. We show that the strategy of the access point can be implemented by a simple on-line scheduling policy. A surprising result is that the channel reliabilities need not be known a priori.

Key Words: Delays, Real-Time Networks, Wireless Networks, Admission Control, Scheduling, Utility Maximization, Network Utility Maximization.

1 Introduction

We begin by describing the context which motivates the formulation of the problem and the theory that we develop in this paper. There is an increasing demand for using wireless networks to serve real-time flows which require strict delay bounds on each packet. Applications that generate such flows include Voice-over-IP (VoIP), video streaming, on-line gaming, networked control, etc. In these applications, packets are only useful if they are delivered within a hard delay bound. Otherwise, the packets "expire," and are not useful. In this paper, we propose to measure the performance of a flow by its *timely-throughput*, defined as the throughput of packets that are delivered before their delay bounds. We assume that each flow has a specified timely-throughput requirement, and study both the specific problem of how to provide guarantees on fulfilling the requirements of all flows in the system, as well as a variation where the flows are elastic and the goal is to maximize a utility function of the timely throughputs. Serving such real-time flows in wireless networks is particularly challenging. Since wireless signals are subject to shadowing, fading, and interference, wireless transmissions are usually unreliable. Therefore, when formulating the problem of serving real-time flows in wireless networks, it is necessary to explicitly take into account the unreliable nature of wireless transmissions.

In this paper, we propose a framework for studying real-time wireless networking. There are two particular challenges in advancing such a framework for further study. First, the criteria that are studied should be relevant for real-time services. Second, the model should both capture the relevant constraints of the wireless medium and should also yield tractable results that can be implemented. So motivated, we propose a framework that models the application specifications as jointly including a per-packet delay bound and timely-throughput requirement, and on the wireless side models the unreliable nature of transmissions. It turns out that this overall model allows the development of a rich theory with interesting, and often, to us at least, surprising, results. Within this framework we will study two problems, one concerning feasi-

bility of inelastic flows in a simple context [1], and another concerning optimization of elastic flows [2]. There are several other results concerning these models that we will not address here. We refer the reader to [3–6] for these results. Reference [3] addresses the problem of more general arrivals, [4] addresses the problem of more general models for wireless fading and rate adaptation, [5] incorporates rate adaptation for the optimization of elastic flows and addresses selfish behaviors of clients, and [6] considers the problem of broadcasting real-time flows.

We begin by studying the problem of characterizing when it is feasible to fulfill a given set of timely-throughput requirements of the clients, given the channel unreliabilities. We establish a necessary and sufficient condition for feasibility [1]. This condition involves exponentially many tests. We subsequently show that the number of tests can be greatly reduced and develop a polynomial-time, actually nearly linear time, algorithm for determining feasibility [1]. This algorithm solves the admission control problem.

We next study the problem of packet scheduling [1]. We propose two simple on-line scheduling policies. We prove that these scheduling policies fulfill the timely-throughput requirements of all flows in the system, as long as the system as a whole is feasible.

Next, we relax the assumption that flows specify their timely-throughput requirements, and discuss how to allocate a timely-throughput to each flow [2]. That is, we consider the problem of elastic flows. The goal is to provide levels of feasible timely-throughputs for the flows that maximize the total utility of the system. This optimization problem can be decomposed into two subproblems, as well known from earlier work, see [7–9], and this decomposition can be naturally interpreted as a bidding game. This bidding game involves each flow selfishly deciding its own payment, while the centralized scheduler, or the access point, is responsible for choosing a suitable scheduling policy to achieve weighted proportional fairness among flows. We show that there exists a simple, and somewhat surprising, on-line scheduling policy that achieves weighted proportional fairness among flows [2]. This policy does not require anyone, neither the clients nor the central scheduler, to know the channel reliabilities.

In summary, our contributions include the following. First, we propose a model for a wireless system serving multiple real-time flows. We then provide solutions for three important problems, namely, admission control, packet scheduling, and utility maximization. We also provide simulations that illustrate the theoretical results.

The rest of the paper is organized as follows: Section 2 summarizes some related work

dealing with serving real-time flows in wireless systems. Section 3 formally introduces the model. We then derive a condition for feasibility in Section 4. In Section 5, the problem of packet scheduling is addressed. In Section 6, we propose an efficient algorithm for admission control. In Section 7, we discuss the problem of utility maximization and propose a bidding game to achieve the maximum utility. The strategy for the AP in the bidding game is further studied in Section 8, and an on-line scheduling policy for the AP is proposed. The optimality of this on-line scheduling policy is established in Section 9. Section 10 provides simulation results. Finally, Section 11 concludes this paper.

2 Related Work

There has been much work on the three important problems in computer networks: admission control, packet scheduling, and utility maximization. Xiao et al [10] and Pong et al [11] have proposed admission control algorithms for scenarios where the performance of each client solely depends on its allocated bandwidth, but do not consider any guarantees on delay bounds. Garg et al [12], Zhai et al [13], and Shin and Schulzrinne [14] have used various performance statistics to predict feasibility, but do not consider theoretical problems underlying their proposed methods. Gao, Cai, and Ngan [15], Niyato and Hossain [16], and Ahmed [17] have surveyed existing work on admission control for various wireless systems.

Tassiulas and Ephremides [18] have proposed a scheduling policy for networks with time-varying connectivity and proved that the policy is throughput optimal. Neely [19] has further shown that this policy achieves constant average delay. Johnsson and Cox [20] have proposed a policy to achieve both small packet delay and high throughput. Dua and Bambos [21] have studied the tradeoff between user fairness and system performance. These works also do not provide any theoretical guarantees. Raghunathan et al [22] and Shakkottai and Srikant [23] have proposed policies that are proved to minimize the total number of expired packets in the system, but do not provide guarantees on the individual performance of each client in the system. Fattah and Leung [24] and Cao and Li [25] have surveyed works on scheduling policies for wireless systems.

Kelly [7] and Kelly, Maulloo, and Tan [8] have considered the problem of utility maximization in wireline networks, and have proposed a decomposition technique for this problem. Lin and Shroff [26] have also studied utility maximization by taking multi-path routing into ac-

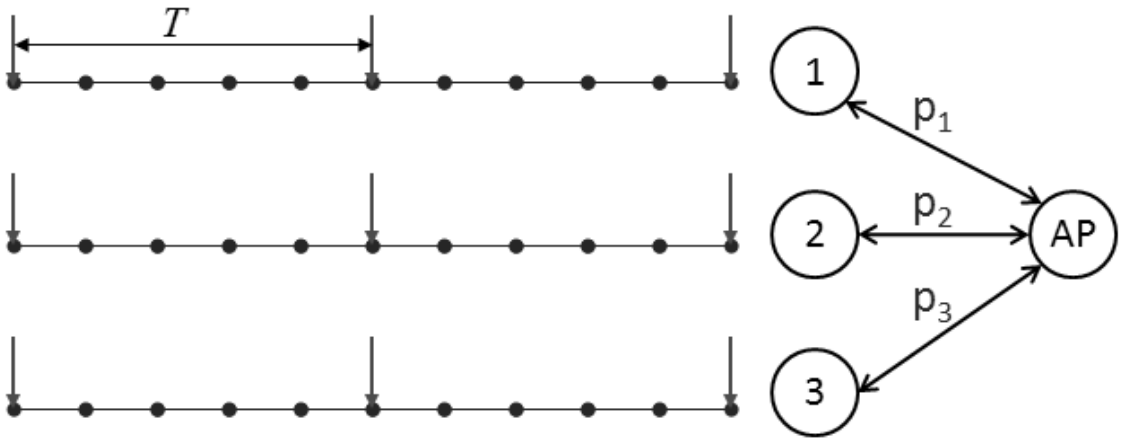


Figure 1: An example that illustrates the system. The right half of the figure illustrates the single server, called access point(AP) serving multiple customer streams, called clients. The service time of a customer from client c is geometrically distributed with mean $1/p_c$, which captures an unreliable channel with probability p_c of success between AP and client c. The left half of the figure shows the timeline of each client's periodic arrival process, where each arrow indicates the arrival of a packet.

count. Xiao, Shroff, and Chong [27], Cao and Li [28], and Bianchi, Campbell, and Liao [29] have studied the problem of resource allocation in wireless networks to either achieve maximum total utility or fairness among all clients in the system, but do not consider delay guarantees.

3 Model

We consider a system with C wireless clients, numbered as $1, 2, \ldots, C$, and one access point (AP); see Figure 1. Each client is associated with a real-time flow that generates traffic that requires delay guarantees. The AP is in charge of scheduling all transmissions in the system.

We assume that time is slotted. The duration of a time slot is set to be able to accommodate the time needed for a transmission between the AP and a client. Time slots are numbered as $\{1, 2, 3, \ldots\}$, and are further grouped into *intervals*, where an interval consists of T consecutive time slots in $(kT, (k+1)T]$, for each k. Each client generates one packet at the beginning of each interval. We assume that each client has a delay bound of T time slots for each of its packets. That is, packets that are generated at the beginning of an interval are required to be delivered before the end of the interval, if they are to be useful. If a packet is not delivered by the end of the interval that it is generated, the packet expires and is removed from the system.

By removing expired packets, the delays of all delivered packets are guaranteed to be at most T time slots.

We consider unreliable and heterogeneous wireless channels. When the AP schedules a transmission for a client c, the transmission is successful with probability p_c, and the transmission fails, either due to channel fading or packet collision, with probability $1 - p_c$.

Since wireless transmissions are unreliable, it may not be possible to deliver all packets before their respective delay bounds. Thus, we measure the performance of a client by its *timely-throughput*, which is defined as the long-term average number of packets that are delivered for the client per interval. To be more specific, let $e_c(k)$ be the indicator function that a packet is delivered for client c in the k^{th} interval. The timely-throughput of client c is then defined as $\liminf_{K\to\infty} \frac{\sum_{k=1}^{K} e_c(k)}{K}$.

Suppose now that each client c has a minimum timely-throughput requirement, q_c. A scheduling policy is one that decides which client to transmit in each time slot, based on the history of all the information concerning the system up to that time slot.

Definition 1. *A system is* fulfilled *by a scheduling policy η if, under η, the timely-throughput of each client is at least as large as its requirement, i.e., $\liminf_{K\to\infty} \frac{\sum_{k=1}^{K} e_c(k)}{K} \geq q_c$, with probability one.*

This paper studies the problem of determining whether it is feasible to fulfill a system, and how to actually fulfill a feasible system.

Definition 2. *The system is said to be* feasible *if there exists some scheduling policy η that fulfills it.*

Definition 3. *A scheduling policy η is* feasibility optimal *if it fulfills every feasible system.*

From a queueing system perspective, this system can be regarded as a discrete-time system with C customer classes and one server. Customers of each class arrive periodically with period T, and in a synchronized way. That is, one customer arrives for each class at times $t = 0, T, 2T, 3T, \ldots$. The service time of customers of class c is geometrically distributed with mean $1/p_c$. The server can serve at most one customer at each time t. Customers balk when their waiting time in the system exceeds T. The service can be preemptive. The server can make its decision on which customer to serve in a slot t based on the entire history of the system up to time t. The timely throughput of class c is the rate at which customers of class c leave after completing service. Each class c requires a minimum timely-throughput q_c. Given

$\{(p_c, q_c, T) : 1 \leq c \leq C\}$, we would like to determine whether it is feasible, and, if so, we would like to determine a scheduling policy that provides at least the desired minimum timely throughput to each client. Looking ahead, in Section 7, we will subsequently also consider the problem of utility maximization, where each client has a utility function $U_c(q_c)$ that is strictly increasing and strictly concave. Then the goal will be to maximize the total system utility $\sum_{c=1}^{C} U_c(q_c)$, over the set of feasible vectors $(q_1, q_2, \ldots, q_C)$ of timely-throughputs.

4 A Necessary Condition for Feasibility

We now analyze the problem of determining whether a system is feasible. We will establish a necessary condition for feasibility, which will be constructively shown to be also sufficient for feasibility in Section 5.

The timely-throughput of a client depends only on the proportion of time slots that that client is transmitted. This is formally captured by the following lemma.

Lemma 1. *There exists a policy that yields for each client c a timely throughput of at least q_c with probability one if and only if the long-term average number of time slots per interval that the AP spends transmitting packets for client c under η is at least $\frac{q_c}{p_c}$ with probability one.*

Proof. Define:

$$u_c(t) = \begin{cases} 1, & \text{if client } c \text{ transmits at time } t, \\ 0, & \text{otherwise,} \end{cases}$$

and

$$d_c(t) = \begin{cases} 1, & \text{if a packet delivered for client } c \text{ at time } t, \\ 0, & \text{otherwise.} \end{cases}$$

Let $\mathfrak{F}_t$ be the σ-algebra generated by all events in the system prior to time t. Then $E[d_c(t)|\mathfrak{F}_{t-1}, u_c(t)] = p_c u_c(t)$. Hence, by the martingale stability theorem of Loeve [30],

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} [d_c(t) - p_c u_c(t)] = 0, \text{ a.s.} \tag{1}$$

Therefore, we have,

$$\liminf_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} d_c(t) \geq \frac{q_c}{T}, \text{ a.s.} \Leftrightarrow \liminf_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} u_c(t) \geq \frac{q_c}{p_c T}, \text{ a.s.}$$

□

We hereafter denote $w_c := \frac{q_c}{p_c}$ as the *work load* of client c. Evaluating whether a system is fulfilled by a policy η is then equivalent to evaluating whether the average number of time slots per interval that the AP schedules transmissions for each client c is at least w_c. Since the AP can schedule at most T transmissions in an interval, we can immediately obtain the following necessary condition for feasibility.

Lemma 2. *A system is feasible only if* $\sum_{c=1}^{N} w_c \leq T$.

However, and this is where the problem begins to become interesting, the above condition is not sufficient. To understand this, consider the example illustrated in Figure 2. There are two clients, so $C = 2$, and the period is $T = 3$. Suppose that in the first slot in an interval, client 1 transmits. Suppose that the transmission is indeed successful, which occurs with probability p_1. Then since there is only one more packet in the system, namely that of client 2, it transmits in the second slot. With probability p_2, it too is successful. In that case, the last slot in the interval necessary has to remain idle, since there are no more packets in the system. Thus the expected number of idle slots in an interval is at least $p_1 p_2$. Hence the system can be busy on average for at most $3 - p_1 p_2$ slots in each interval. Thus, it is necessary that $\sum_{c=1}^{2} w_c \leq 3 - p_1 p_2$, in order that (q_1, q_2) be feasible. Therefore the above condition shown in Lemma 2 is not sufficient.

Thus, the expected number of time slots that will be forced to be idle also needs to be considered when evaluating feasibility. In general, the expected number of idle time slots depends on the scheduling policy used; however, we show that it is the same for all *work conserving* policies.

Definition 4. *A scheduling policy is said to be* work conserving *if it schedules some packet for transmission that has neither expired nor been delivered, whenever there is some such packet in the system, and only idles a slot when there are no such packets waiting to be transmitted.*

Lemma 3. *The expected number of idle time slots in an interval is the same for all work conserving policies.*

Proof. Let γ_c be a geometrically distributed random variable with mean $1/p_c$; it is the random number of transmissions that the AP needs to schedule, in order to deliver a packet for client c in some interval. Since work conserving policies only idle when all packets generated in an interval have been delivered, the number of idle time slots in an interval can then be expressed as $\max(0, T - \sum_{c=1}^{C} \gamma_c) = (T - \sum_{c=1}^{C} \gamma_c)^+$, which is the same for all work conserving policies.

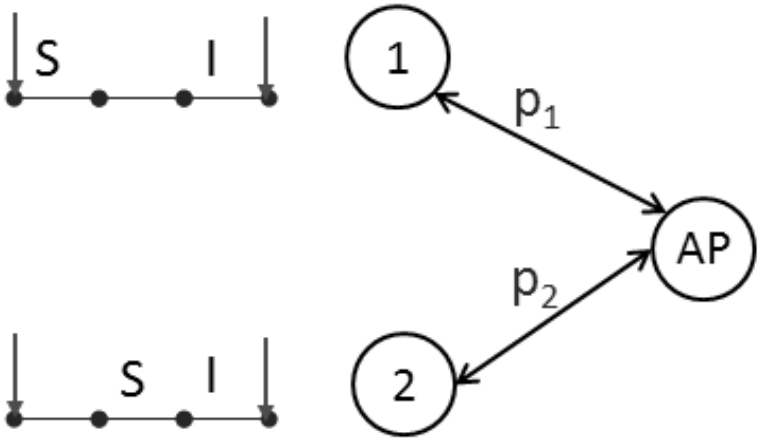


Figure 2: An illustration of the example in Section 4. We use 'S' to denote a successful transmission, and 'I' to denote an idle time slot.

Therefore, the expected number of idle time slots in an interval, $E\{(T - \sum_{c=1}^{C} \gamma_c)^+\}$, is also the same under all work conserving policies. □

The following lemma allows us to hereafter focus on work conserving policies.

Lemma 4. *For any feasible system, there exists a work conserving policy that fulfills it.*

Proof. Consider any feasible system. By definition, it can be fulfilled by some scheduling policy η'. We construct a work conserving policy η from η', in two steps. First, let η'' be a policy that moves all idle slots to the end of the period by taking the same action that η' would have taken after a contiguous sequence of idle slots. We can stochastically couple the behavior under η' and η''. Next, we construct the policy η by scheduling transmissions the same way as η'' except that η randomly picks a packet that has not been delivered for transmission in each time slot that η'' chooses to idle, whenever such packet exists. Thus, η is a work conserving policy. Moreover, it stochastically delivers more packets for each client than η'. Thus, η also fulfills the system. □

Let us denote by $I_{\{1,2,\ldots,C\}}$ the expected number of idle time slots in an interval under a work conserving policy; noting that it is the same under all work conserving policies.

Clearly, we can strengthen the necessary condition of Lemma 2 as follows.

Lemma 5. *A system is feasible only if* $\sum_{c=1}^{C} w_c + I_{\{1,2,\ldots,C\}} \leq T$.

Now we again ask the question. Is this a sufficient condition? Surprisingly, the answer is again no. To understand this, we again present a counterexample.

Example 1. *Consider a system with interval length $T = 3$, and two clients. The reliabilities for both clients are $p_1 = p_2 = 0.5$. Client 1 requires a timely throughput of $q_1 = 0.876$, while the timely throughput requirement of client 2 is $q_2 = 0.45$.*

Now, we have:

$$w_1 = 1.76,$$
$$w_2 = 0.9,$$
$$I_{\{1\}} = I_{\{2\}} = 1.25,$$
$$I_{\{1,2\}} = 0.25.$$

If we evaluate the condition for the subset of $S = \{1\}$, we find $w_1 = 1.76 > 1.75 = T - I_{\{1\}}$. This indicates that the system is not feasible even if only client 1 were present. (However, if we evaluate the condition for all clients $\{1, 2\}$, we have $w_1 + w_2 = 2.66 < 2.75 = T - I_{\{1,2\}}$.) Thus, this example suggests that merely evaluating the condition for the set of all clients is not sufficient. □

From this example, it is clear that we will need to consider all subsets of the set of clients. Accordingly, let us fix a subset $S \subseteq \{1, 2, \ldots, C\}$, and consider a hypothetical problem where only the clients in the subset exist and need to be served. Let us denote by I_S the expected number of idle time slots in an interval under any work conserving scheduling policy when only these clients in the subset $S \subseteq \{1, 2, \ldots, C\}$ are present in the system. The value of I_S can be formally expressed as $E\{T - \sum_{n \in S} \gamma_c)^+\}$, where γ_c is the geometric random variable with mean $1/p_c$ that indicates the number of transmissions that the AP needs to schedule for client c before successfully delivering its packet. We now obtain an even stronger necessary condition for feasibility.

Lemma 6. *A system is feasible only if $\sum_{c \in S} w_c + I_S \leq T$, for all $S \subseteq \{1, 2, \ldots, C\}$.*

One can ask the question: Why is this condition not automatically satisfied whenever the condition of Lemma 5 is satisfied. The answer lies in the observation that while the first term $\sum_{c \in S} w_c$ is monotonically increasing in S, the second term I_S is monotonically decreasing in S. Hence the sum $\sum_{c \in S} w_c + I_S$ is not necessarily monotonically increasing in S. Thus it may not attain its maximum when S is the maximum set $\{1, 2, \ldots, C\}$. This is why checking only the condition of Lemma 5 is not sufficient.

It turns out that the condition of Lemma 6 is indeed necessary and sufficient, as we prove in Section 5. Our proof will be constructive, and so we now turn to the issue of scheduling policies.

5 Scheduling Policies

In this section, we describe two scheduling policies and prove that they both fulfill every system that satisfies the condition for feasibility in Lemma 6. These two policies are hence both feasibility optimal.

Both policies are what we call as *largest debt first policies*. Under a largest debt first policy, the AP computes the *debt* that it owes to each client at the beginning of each interval, where the particular definition of debt is specified by the corresponding policy. The AP then prioritizes all clients according to their debts. In each time slot in the interval, the AP schedules the transmission for an undelivered packet from the client which has the largest debt among all clients whose packets have not yet been delivered in the interval.

We propose two different definitions for debt, and therefore two different scheduling policies. The first type of debt is called the *time-based debt*. It is derived from the work loads of clients. Let $f_c(k)$ be the number of time slots in which the AP schedules transmissions for client c in the k^{th} interval. The time-based debt of client c at the beginning of the $(K+1)^{th}$ interval is then defined as $Kw_c - \sum_{k=1}^{K} f_c(k)$. The rationale is the following. From Lemma 1 we know that client c needs on average Kw_c slots in K intervals, in order to fulfill the needs of client c. Thus Kw_c is the number of slots that the AP *owes* client c. However, if it has actually spent less time catering to client c, then the balance between what it owes and what is has spent on client c is the time-based debt that it owe to client c.

The other type of debt is called the *weighted-delivery debt*. Let $e_c(k)$ be the indicator function that the AP actually delivers a packet for client c in the k^{th} interval. The weighted-delivery debt of client c at the beginning of the $(K+1)^{th}$ interval is defined as $(Kq_c - \sum_{k=1}^{K} e_c(k))/p_c$. Again, the first term Kq_c represents the number of packets that should have been delivered to client c in the first K intervals, on average, and thus represents what is owed to client c. The second term, $\sum_{k=1}^{K} e_c(k)$, is the actual number of packets delivered, and so the balance is the *delivery debt* it owe to client c. The normalization by dividing by p_c is therefore called the weighted-delivered debt. It may be noted that the weight $1/p_c$ represents the number of slots that is needed to deliver a packet on average. Hence, this normalization is actually converting packets into an

equivalent number of time slots.

The resulting largest debt first policies are called the *largest time-based debt first policy* and the *largest weighted-delivery debt first policy*, respectively.

5.1 Proof of Feasibility Optimality of the Two Largest Debt First Policies

We will prove that both largest debt first policies are feasibility optimal. The proofs are based on the Blackwell's approachability theorem [31]. We first introduce this theorem.

Consider a single-player repeated game. In each round i of the game, the player chooses an action, $a(i) \in A$, based on events happening prior to this round. The player then receives a C-dimensional *vector* of reward, denoted by $v(i) \in \mathbb{R}^N$, at random. The distribution of $v(i)$ only depends on the action taken by the player, $a(i)$. Blackwell studied the long-term average payoff vector, $\lim_{j\to\infty} \frac{\sum_{i=1}^{j} v(i)}{j}$, and introduced the concept of *approachability*. (This is done by Blackwell in a more general zero-sum game context than what we need here).

Definition 5. *Let $B \subseteq \mathbb{R}^C$ be a measurable set in C-dimensional space. B is said to be approachable under a policy η if, under η, the distance between B and $\frac{\sum_{i=1}^{j} v(i)}{j}$, which is the average payoff up to round j, converges to 0 almost surely as j goes to infinity.*

Blackwell then provided a sufficient condition for approachability.

Theorem 1 ([31])**.** *Let $B \subseteq \mathbb{R}^C$ be a closed set in C-dimensional space. For each point $x \notin B$, denote $y(x)$ as the closest point in B to x, and $h(x)$ as the hyperplane perpendicular to the line segment $xy(x)$ that passes through $y(x)$. Consider a policy η that has the following property: Whenever $\frac{\sum_{i=1}^{j} v(i)}{j}$ is not in B, η chooses an action a for round $j+1$ whose expected payoff, $E\{v(j+1)|a(j+1) = a\}$, is separated from $\frac{\sum_{i=1}^{j} v(i)}{j}$ by the hyperplane $h(\frac{\sum_{i=1}^{j} v(i)}{j})$. Then B is approachable under the policy η.*

We now prove that the two largest debt first policies are feasibility optimal. In the following proof, and throughout the rest of the paper, we use $[v_c]$ to denote the vector consisting of $[v_1, v_2, \ldots, v_C]$.

Theorem 2. *The largest time-based debt first policy is feasibility optimal.*

Proof. We prove that the largest time-based debt first policy fulfills every system that satisfies the necessary condition for feasibility in Lemma 6.

Let $f_c(k)$ be the number of time slots in which the AP schedules transmissions for client c in the k^{th} interval. The time-based debt of client c at the beginning of the $(k+1)^{th}$ interval is then $d_c(k) := kw_c - \sum_{j=1}^{k} f_c(j)$. Lemma 1 has shown that a system is fulfilled if $\limsup_{k\to\infty} \frac{d_c(k)}{k} \leq 0$. We now show that, under the largest time-based debt first policy, the vector $[\frac{d_c(k)}{k}]$ does indeed approach the set $B = \{v = [v_c] \in \mathbb{R}^C | v_c \leq 0, \forall c\}$.

We apply Theorem 1, where the AP acts as the player in the repeated game, whose actions are scheduling decisions. The payoff of the AP in the k^{th} interval is the vector $[w_c - f_c(k)]$. Assume that, at the beginning of the $(k+1)^{th}$ interval, the vector $[d_c(k)]$ is not in B. Without loss of generality, assume that $d_1(k) \geq d_2(k) \geq \cdots \geq d_m(k) > 0 \geq d_{m+1}(k) \geq \ldots$. The closest point in B to $[d_c(k)]$ is $[0, 0, \ldots, 0, d_{m+1}(k), d_{m+2}(k), \ldots]$. The hyperplane $h([d_c(k)])$ is then $\{x \in \mathbb{R}^C | g(x) := \sum_{c=1}^{m} d_c(k)x = 0\}$.

For each l, the largest time-based debt first policy gives higher priority to clients 1 through l than the rest of clients in the $(k+1)^{th}$ interval. Thus, we have $\sum_{c=1}^{l} E\{f_c(k+1)\} = T - I_{\{1,\ldots,l\}}$, where $I_{\{1,\ldots,l\}}$ is the expected number of idle time slots in an interval when the AP only schedules transmissions for clients 1 through l. We also have $E\{f_l(k+1)\} = I_{\{1,\ldots,l-1\}} - I_{\{1,\ldots,l\}}$, where $I_\phi = T$. The expected payoff in the $(k+1)^{th}$ round is then $[w_c - (I_{\{1,\ldots,c-1\}} - I_{\{1,\ldots,c\}})]$. Thus,

$$\begin{aligned} g([w_c - (I_{\{1,\ldots,c-1\}} - I_{\{1,\ldots,c\}})]) &= \sum_{c=1}^{m} d_c(k)(w_c - (I_{\{1,\ldots,c-1\}} - I_{\{1,\ldots,c\}})) \\ &= \sum_{c=1}^{m-1} [d_c(k) - d_{c+1}(k)](\sum_{l=1}^{c} w_l + I_{\{1,\ldots,c\}} - T) \\ &\quad + d_m(k)(\sum_{l=1}^{m} w_l + I_{\{1,\ldots,m\}} - T) \\ &\leq 0. \end{aligned}$$

Lemma 6 has shown that every feasible system has $\sum_{l=1}^{c} w_l + I_{\{1,\ldots,c\}} - T \leq 0$, for all c. Thus, the last inequality holds since $d_1(k) \geq d_2(k) \cdots \geq d_m(k) > 0$. On the other hand, $g([d_c(k)]) = \sum_{c=1}^{m} d_c(k)^2 > 0$. Thus, $h([d_c(k)]$ separates $[d_c(k)]$ and $[w_c - (I_{\{1,\ldots,c-1\}} - I_{\{1,\ldots,c\}})]$. By Theorem 1, $[\frac{d_c(k)}{k}]$ approaches the set B under the largest time-based debt policy. □

Theorem 3. *The largest weighted-delivery debt first policy is feasibility optimal.*

Proof. We prove that the largest weighted-delivery debt first policy fulfills every system that satisfies the necessary condition for feasibility in Lemma 6.

Let $e_c(k)$ be the indicator function that the AP delivers a packet for client c in the k^{th} interval. The weighted-delivery debt of client c at the beginning of the $(k+1)^{th}$ interval is defined as $d_c(k) := (kq_c - \sum_{j=1}^{k} e_c(j))/p_c$. Obviously, a system is fulfilled if the C-dimensional vector $[\frac{d_c(k)}{k}]$ approaches $B = \{v = [v_c] \in \mathbb{R}^C | v_c \leq 0, \forall c\}$.

Similar to the proof of Theorem 2, we again apply Theorem 1. The AP acts as the player, whose action is the scheduling decision, and the payoff is the vector $[(q_c - e_c(k))/p_c]$ in each interval k. Suppose that the vector $[d_c(k)]$ is not in B at the beginning of the $(k+1)^{th}$ interval. Without loss of generality, assume that $d_1(k) \geq d_2(k) \geq \cdots \geq d_m(k) > 0 \geq d_{m+1}(k) \geq \ldots$. The closest point in B to $[d_c(k)]$ is $[0, 0, \ldots, 0, d_{m+1}(k), d_{m+2}(k), \ldots]$. The hyperplane $h([d_c(k)])$ is then $\{x \in \mathbb{R}^C | g(x) := \sum_{c=1}^{m} d_c(k)x = 0\}$.

The largest weighted-delivery debt first policy prioritizes according to the order $1, 2, \ldots$ in the $(k+1)^{th}$ interval. As shown in the proof of Theorem 2, the expected amount of time that the AP spends transmitting the packet for client c in the $(k+1)^{th}$ interval is $(I_{1,2,\ldots,c-1} - I_{1,2,\ldots,c})$. By Lemma 1, $E\{e_c(k)\} = p_c(I_{1,2,\ldots,c-1} - I_{1,2,\ldots,c})$, and the expected payoff in the $(k+1)^{th}$ interval is $[(q_c - p_c(I_{1,2,\ldots,c-1} - I_{1,2,\ldots,c}))/p_c] = [w_c - (I_{1,2,\ldots,c-1} - I_{1,2,\ldots,c})]$. As shown in the proof of Theorem 2, $g([w_c - (I_{1,2,\ldots,c-1} - I_{1,2,\ldots,c})]) \leq 0$. On the other hand, $g([d_c(k)]) = \sum_{c=1}^{m} d_c(k)^2 > 0$. Thus, $h([d_c(k)])$ separates $[d_c(k)]$ and $[(q_c - p_c(I_{1,2,\ldots,c-1} - I_{1,2,\ldots,c}))/p_c]$. By Theorem 1, $[\frac{d_c(k)}{k}]$ approaches B as $k \to \infty$. □

Finally, we note that in the proofs of Theorem 2 and Theorem 3, both largest debt first policies fulfill every system that satisfies the necessary condition in Lemma 6. Thus, this condition is also sufficient.

Theorem 4. *A system is feasible if and only if $\sum_{c \in S} w_c + I_S \leq T$, for all $S \subseteq \{1, 2, \ldots, C\}$.*

6 An Efficient Algorithm for Admission Control

Theorem 4 has established a necessary and sufficient condition for a system to be feasible. Thus, to execute admission control, one only needs to evaluate if this necessary and sufficient condition is satisfied. However, this condition consists of checking the inequality $\sum_{c \in S} w_c + I_S \leq T$ for every subset $S \subseteq \{1, 2, \ldots, C\}$ of clients, making it a computationally complex test, since there are 2^C such subsets to check. In this section, we show that the condition can be greatly simplified. We derive a polynomial-time algorithm, actually a nearly linear time algorithm, for

admission control. The algorithm is based on the following theorem.

Theorem 5. *Order the clients so that $q_1 \geq q_2 \geq \cdots \geq q_C$. Let S_k be the subset $\{1, 2, \ldots, k\}$. The system is then feasible if and only if $\sum_{c \in S_k} w_c + I_{S_k} \leq T$, for all $1 \leq k \leq C$.*

Proof. It is obvious that the above condition is necessary for feasibility. We only need to show that it is also sufficient.

Consider any infeasible system. Define a *minimal infeasible set* of this system as a subset S for which $\sum_{c \in S} w_c + I_S > T$, but for each $S' \subsetneq S$, $\sum_{c \in S'} w_c + I_{S'} \leq T$. For every infeasible system, there must exist at least one minimal infeasible set S. Fix S, and let m be the largest element in S; that is, $m = \min\{k | S \subseteq S_k\}$. We prove that $\sum_{c \in S_m} w_c + I_{S_m} > T$.

If $S = S_m$, then we are done. Otherwise, let l be the largest element in $S_m \backslash S$. We want to show that $\sum_{c \in S \cup \{l\}} w_c + I_{S \cup \{l\}} \geq \sum_{c \in S} w_c + I_S$.

Consider a scheduling policy η that only transmits packets for clients in $S \backslash \{m\}$ and fulfills them. Such policy exists because S is a minimal infeasible set, whence the subset $S \backslash \{m\}$ is feasible. We expand η by making it schedule transmissions for client m in an interval only after all packets for clients in $S \backslash \{m\}$ are delivered, and schedule transmissions for client l only after the packet for client m is delivered. Under this policy, the expected amount of time that the AP spends transmitting the packet for client l in an interval is $I_S - I_{S \cup \{l\}}$, and so the timely-throughput of client l is $p_l(I_S - I_{S \cup \{l\}})$ by Lemma 1. Also, the timely-throughput of client m is strictly less than q_m, since the set S is infeasible, and all clients in $S \backslash \{m\}$ are fulfilled under this policy. Further, the timely-throughput of client l is no larger than that of client m, as it is scheduled only after the packet for client m is delivered. Thus, $p_l(I_S - I_{S \cup \{l\}}) < q_m$. We now have

$$\begin{aligned}
(\sum_{c \in S \cup \{l\}} w_c + I_{S \cup \{l\}}) - (\sum_{c \in S} w_c + I_S) =& \frac{q_l}{p_l} - (I_S - I_{S \cup \{l\}}) \\
=& [q_l - p_l(I_S - I_{S \cup \{l\}})]/p_l \\
>& (q_l - q_m)/p_l \geq 0,
\end{aligned}$$

where the last inequality holds because $q_l \geq q_m$. Hence, $\sum_{n \in S \cup \{l\}} w_n + I_{S \cup \{l\}} > \sum_{n \in S} w_n + I_S > T$.

If $S \cup \{l\} = S_m$, then we are done. Otherwise, we select l' to be the largest element in $S_m \backslash (S \cup \{l\})$. We expand η by making it schedule transmissions for client m in an interval

only after all packets for clients in $S\backslash\{m\}$ are delivered, and schedule transmissions for client l (or client l') only after the packet for client m (or the packet for client l, respectively) is delivered. Under this policy, the expected number of time slots that the AP spends on client l' is $I_{S\cup\{l\}} - I_{S\cup\{l,l'\}}$, and thus the timely-throughput of client l' is $p_{l'}(I_{S\cup\{l\}} - I_{S\cup\{l,l'\}})$. Further, the timely-throughput of client l' is no larger than that of client l, which in turn is strictly less than q_m. We now have

$$\begin{aligned}(\sum_{c\in S\cup\{l,l'\}} w_c + I_{S\cup\{l,l'\}}) - (\sum_{c\in S\cup\{l\}} w_c + I_{S\cup\{l\}}) =& \frac{q_{l'}}{p_{l'}} - (I_{S\cup\{l\}} - I_{S\cup\{l,l'\}}) \\ =& [q_{l'} - p_{l'}(I_{S\cup\{l\}} - I_{S\cup\{l,l'\}})]/p_{l'} \\ >& (q_{l'} - q_m)/p_{l'} \geq 0,\end{aligned}$$

Hence, $\sum_{c\in S\cup\{l,l'\}} w_c + I_{S\cup\{l,l'\}} > \sum_{c\in S} w_c + I_S > T$.

If $S\cup\{l,l'\} = S_m$, then we are done. Otherwise, we select l'' to be the largest element in $S_m\backslash(S\cup\{l,l'\})$ and repeat the above procedure. By induction, we establish that $\sum_{c\in S_m} w_c + I_{S_m} > T$. □

Thus, we only need to evaluate a total of C inequalities in order to decide whether a system is feasible. Each test, in turn, involves computing I_{S_k}. We now show that there is an efficient algorithm that computes I_{S_k} incrementally. We note that $S_{k+1} = S_k \cup \{k+1\}$. Let $\pi_{S_k}(t)$ be the probability that $\sum_{c\in S_k}\gamma_c = t$, where γ_c is the geometrically distributed random variable with mean $1/p_c$ that represents the number of transmissions that the AP need to schedule for client c in order to deliver the packet of client c. Then, $I_{S_k} = \sum_{t=1}^{T}(T-t)\pi_{S_k}(t))$. Further, $\pi_{S_{k+1}}(t) = \sum_{\tau=1}^{t-1}\pi_{S_k}(\tau)Prob\{\gamma_{k+1} = t-\tau\} = \sum_{\tau=1}^{t-1}\pi_{S_k}(t)p_{k+1}(1-p_{k+1})^{t-\tau-1}$. Thus, the vector $[\pi_{S_{k+1}}(t)|1\leq t\leq T]$ is the convolution of the vectors $[\pi_{S_k}(t)]$ and $[p_{k+1}(1-p_{k+1})^{t-1}]$, which can be computed in $O(T^2)$ time by brute force, or in $O(T\log T)$ by using the Fast Fourier Transform Algorithm [32].

A complete algorithm is shown in Algorithm 1. The complexity of Algorithm 1 is $O(N\log N + NT\log T)$ or $O(N\log N + NT^2)$, depending on the implementation of convolution.

Algorithm 1 IsFeasible

Sort all clients so that $q_1 \geq q_2 \geq \cdots \geq q_C$
$totalW \leftarrow \frac{q_1}{p_1}$
$[\pi_{S_1}(t)] \leftarrow [p_1(1-p_1)^{t-1}]$
$totalI \leftarrow \sum_{t=1}^{T-1}(T-t)\pi_{S_1}(t)$
if $totalW + totalI > T$ **then**
  return Infeasible
end if
for $c = 2$ to C **do**
  $totalW \leftarrow totalW + \frac{q_c}{p_c}$
  $[\pi_{S_c}(t)] \leftarrow [\pi_{S_{c-1}}(t)] * [p_c(1-p_c)^{t-1}]$
  $totalI \leftarrow \sum_{t=1}^{T-1}(T-t)\pi_{S_c}(t)$
  if $totalW + totalI > T$ **then**
    return Infeasible
  end if
end for
return Feasible

7 Utility Maximization for Elastic Traffic

In the previous sections, we have assumed that the timely-throughputs of clients, $[q_c]$, are given and fixed. In applications with elastic traffic, clients do not have rigid timely-throughput requirements. Rather, each client obtains a certain utility based on the timely-throughput that it receives, which reflects the performance of its application when it receives that amount of timely-throughput. In this section and the following one, we address the problem of providing a timely-throughput to each client, so that the sum of the utilities of the clients is maximized. We therefore treat the problem of optimizing real-time traffic with hard deadlines for several clients sharing an AP, under a utility maximization framework.

We assume that each client c has an utility function $U_c(\cdot)$. When the timely-throughput received by client c is q_c, client c obtains an utility of $U_c(q_c)$. We assume that $U_c(q_c)$ is strictly increasing, strictly concave, and continuously differentiable over the range $0 < q_c \leq 1$, with the value at 0 defined as the right limit, possibly $-\infty$. Our goal is to find $[q_c]$ to maximize the total utility in the system, $\sum_{c=1}^{C} U_c(q_c)$, within the set of vectors $[q_c]$ that are feasible. This problem can be formulated as the following optimization problem.

SYSTEM:

$$\text{Max } \sum_{c=1}^{C} U_c(q_c) \tag{2}$$

$$\text{s.t. } \sum_{c \in S} \frac{q_c}{p_c} \leq T - I_S, \forall S \subseteq \{1, 2, \ldots, C\}, \tag{3}$$

$$\text{and } q_c \geq 0, \forall 1 \leq c \leq C. \tag{4}$$

It may be difficult to solve *SYSTEM* directly due to two major problems, one informational and one computational. First, the objective function (2) involves the individual utility function of each client, which may be different from client to client and may not be known to the AP. Second, there are a total of 2^C feasibility constraints in (3), which makes it computationally complex. So, instead of solving *SYSTEM* directly, we adopt a decomposition technique of Kelly [7] , Kelly, Maulloo, Tan [8], and Eisenberg and Gale [9]. Using this approach, we decompose *SYSTEM* into two subproblems, namely, a *CLIENT*$_c$ subproblem and an *ACCESS-POINT* subproblem. The *CLIENT*$_c$ subproblem involves only the utility function of client c, while the *ACCESS-POINT* subproblem considers the system-wide feasibility constraints which couple the clients. We will later show that this decomposition can be interpreted as a bidding game where the behavior of each client is consistent with its own interest. We will show that, surprisingly, there exists a simple on-line scheduling policy that solves the *ACCESS-POINT* subproblem. Moreover, and again surprisingly, this policy does not need explicit knowledge of client channel reliabilities by anybody.

We first describe the Client-AP decomposition. Given a parameter $\psi_c > 0$, representing the "price" per unit of timely-throughput for client c, the *CLIENT*$_c$ subproblem is as follows:

CLIENT$_c$**:** (Given the price $\psi_c > 0$)

$$\text{Max } U_c(\frac{\rho_c}{\psi_c}) - \rho_c \tag{5}$$

$$\text{over } 0 \leq \rho_c \leq \psi_c. \tag{6}$$

On the other hand, given a non-negative vector $[\rho_c]$, representing the "payments" made by the clients, the *ACCESS-POINT*'s subproblem is as follows:

ACCESS-POINT: (Given non-negative payments $[\rho_c]$)

$$\text{Max} \sum_{c=1}^{C} \rho_c \log q_c \tag{7}$$

$$\text{s.t.} \sum_{c \in S} \frac{q_c}{p_c} \leq T - I_S, \forall S \subseteq \{1, 2, \ldots, C\}, \tag{8}$$

$$\text{over } q_c \geq 0, \forall 1 \leq c \leq C. \tag{9}$$

We now prove that the solution of the *SYSTEM* problem follows immediately by jointly solving the *CLIENT*$_c$ subproblem and the *ACCESS-POINT* subproblem. This result is not new; it can be found in [7, 8].

Theorem 6. *There exist non-negative vectors* $[q_c]$, $[\rho_c]$, *and* $[\psi_c]$, *with* $\rho_c = \psi_c q_c$, *with the following properties.*

(i) For c *such that* $\psi_c > 0$, ρ_c *is a solution to* CLIENT$_c$*;*

(ii) Given that each client c *pays* ρ_c *per period,* $[q_c]$ *is a solution to* ACCESS-POINT.

Further, if $[q_c]$, $[\rho_c]$, *and* $[\psi_c]$ *are all positive vectors, the vector* $[q_c]$ *is also a solution to* SYSTEM.

Proof. This proof resembles the one used in [7]. We will first show the existence of $[q_c]$, $[\rho_c]$, and $[\psi_c]$ that satisfy (i) and (ii). We will then show that the resulting $[q_c]$ is also the solution to *SYSTEM*.

There exists some $\epsilon > 0$ so that by letting $q_c \equiv \epsilon$, the vector $[q_c]$ is an interior point of the feasible region for both *SYSTEM* (3) (4), and *ACCESS-POINT* (8) (9). Also, by setting $\rho_c \equiv \epsilon$, ρ_c is also an interior point of the feasible region for *CLIENT*$_n$ (6). Therefore, by Slater's condition, a feasible point for *SYSTEM*, *CLIENT*$_c$, or *ACCESS-POINT*, is the optimal solution for the respective problem if and only if it satisfies the corresponding Karush-Kuhn-Tucker (KKT) condition for the problem. Further, since the feasible region for each of the problems is compact, and the utilities are continuous on it, or since the utility converges to $-\infty$ at $q_c = 0$, there exists an optimal solution to each of them.

The Lagrangian of *SYSTEM* is:

$$L_{SYS}(q,[\lambda_S : S \subseteq \{1,2,\ldots,C\}],[\nu_c]) := -\sum_{c=1}^{C} U_c(q_c) + \sum_{S\subseteq\{1,2,\ldots,C\}} \lambda_S[\sum_{c\in S}\frac{q_c}{p_c} - (T - I_S)] - \sum_{c=1}^{C}\nu_c q_c,$$

where the vectors $[\lambda_S : S \subseteq \{1,2,\ldots,C\}]$ and $[\nu_c]$ are the Lagrange multipliers. By the KKT condition, a vector $[q_c^*]$ is the optimal solution to *SYSTEM* if $[q_c^*]$ is feasible and there exists vectors $[\lambda_S^*]$ and $[\nu_c^*]$ such that:

$$\left.\frac{\partial L_{SYS}}{\partial q_c}\right|_{[q_c^*],[\lambda_S^*],[\nu_S^*]} = -U_c'(q_c^*) + \frac{\sum_{S\ni c}\lambda_S^*}{p_c} - \nu_c^* = 0, \forall 1 \le c \le C, \tag{10}$$

$$\lambda_S^*[\sum_{c\in S}\frac{q_c^*}{p_c} - (T - I_S)] = 0, \forall S \subseteq \{1,2,\ldots,C\}, \tag{11}$$

$$\nu_c^* q_c^* = 0, \forall 1 \le c \le C, \tag{12}$$

$$\lambda_S^* \ge 0, \forall S \subseteq \{1,\ldots,C\}, \text{ and } \nu_c^* \ge 0, \forall 1 \le c \le C. \tag{13}$$

The Lagrangian of *CLIENT*$_c$ is:

$$L_{CLI}(\rho_c,\xi_c) := -U_c(\frac{\rho_c}{\psi_c}) + \rho_c - \xi_c\rho_c,$$

where ξ_c is the Lagrange multiplier for *CLIENT*$_c$. By the KKT condition, ρ_c^* is the optimal solution to *CLIENT*$_c$ if $\rho_c^* \ge 0$ and there exists ξ_c^* such that:

$$\left.\frac{dL_{CLI}}{d\rho_c}\right|_{\rho_c^*,\xi_c^*} = -\frac{1}{\psi_c}U_c'(\frac{\rho_c^*}{\psi_c}) + 1 - \xi_c^* = 0, \tag{14}$$

$$\xi_c^*\rho_c^* = 0, \tag{15}$$

$$\xi_c^* \ge 0. \tag{16}$$

Finally, the Lagrangian of *ACCESS-POINT* is:

$$L_{AP}([q_c],[\zeta_S],[\mu_c]) := -\sum_{c=1}^{C}\rho_c\log q_c + \sum_{S\subseteq\{1,2,\ldots,C\}}\zeta_S[\sum_{c\in S}\frac{q_c}{p_c} - (T - I_S)] - \sum_{c=1}^{C}\mu_c q_c,$$

where $[\zeta_S]$ and $[\mu_c]$ are the Lagrange multipliers. Again, by the KKT condition, a vector $q^* := [q_c^*]$ is the optimal solution to *ACCESS-POINT* if $[q_c^*]$ is feasible and there exist vectors $[\zeta_S^*]$ and $[\mu_c^*]$ such that:

$$\begin{aligned} \left.\frac{\partial L_{AP}}{\partial q_c}\right|_{[q_c^*],[\zeta_S^*],[\mu_c^*]} &= -\frac{\rho_c}{q_c^*} + \frac{\sum_{S \ni c} \zeta_S^*}{p_c} - \mu_c^* \\ &= 0, \forall 1 \le c \le C, \end{aligned} \tag{17}$$

$$\zeta_S^*[\sum_{c \in S} \frac{q_c^*}{p_c} - (T - I_S)] = 0, \forall S \subseteq \{1, 2, \ldots, C\}, \tag{18}$$

$$\mu_c^* q_c^* = 0, \forall 1 \le c \le C, \tag{19}$$

$$\zeta_S^* \ge 0, \forall S \subseteq \{1, \ldots, C\}, \text{ and } \mu_c^* \ge 0, \forall 1 \le c \le C. \tag{20}$$

Assume that $[q_c]$, $[\rho_c]$, and $[\psi_c]$ satisfy (i) and (ii). We wish to show that $[q_c]$ is a solution to $SYSTEM$. Let ξ_c be the Lagrange multiplier for $CLIENT_c$. Since we assume $\psi_c > 0$ for all c, the problem $CLIENT_c$ is well-defined for all c, and so is ξ_c. Also, let $[\zeta_S]$ and $[\mu_c]$ be the Lagrange multipliers for $ACCESS\text{-}POINT$. Since $q_c > 0$, $\mu_c = 0$ by (19). Further, by (17), $\psi_c = \frac{\rho_c}{q_c} = \frac{\sum_{S \ni c} \zeta_S}{p_c}$.

Let $\lambda_S = \zeta_S$ for all S, and $\nu_c = \psi_c \xi_c$ for all c. Since $[q_c]$ is a solution to $ACCESS\text{-}POINT$, it is feasible. We check the KKT condition for *SYSTEM* (10)–(13) and obtain that

$$\begin{aligned} \left.\frac{\partial L_{SYS}}{\partial q_c}\right|_{[q_c],[\lambda_S],[\nu_c]} &= -U_c'(q_c) + \frac{\sum_{S \ni c} \lambda_S}{p_c} - \nu_c \\ &= -U_c'(\tfrac{\rho_c}{\psi_c}) + \psi_c - \psi_c \xi_c = 0, \forall c, \text{ by (14)}, \end{aligned}$$

$$\begin{aligned} \lambda_S[\textstyle\sum_{c \in S} \frac{q_c}{p_c} - (T - I_S)] &= \zeta_S[\textstyle\sum_{c \in S} \frac{q_c}{p_c} - (T - I_S)] \\ &= 0, \forall S, \text{ by (18)}, \end{aligned}$$

$$\nu_c q_c = \xi_c \rho_c = 0, \forall c, \text{ by (15)},$$

$$\lambda_S = \zeta_S \ge 0, \forall S, \text{ by (20)},$$

$$\nu_c = \psi_c \xi_c \ge 0, \forall c, \text{ by (16)}.$$

Therefore, $[q_c]$ solves $SYSTEM$. □

7.1 A Bidding Game

We now show that the decomposition into the *CLIENT*$_c$ subproblem and the *ACCESS-POINT* subproblem can be naturally interpreted and implemented as a bidding game.

We interpret ρ_c as the amount of *payment* that client c pays to the AP per interval. Since Theorem 6 requires that $\rho_c = \psi_c q_c$, ψ_c can be interpreted as the required amount of payment for each unit of timely-throughput, or the *price* per unit of timely-throughput for client c. Under this interpretation, the objective of *CLIENT*$_c$ is to maximize $U_c(\frac{\rho_c}{\psi_c}) - \rho_c = U_c(q_c) - \rho_c$, which is the utility that client c obtains minus its amount of payment. Thus, the *CLIENT*$_c$ subproblem seeks to selfishly maximize the *net utility* of client c, assuming a linear relation between payment and timely-throughput. On the other hand, the objective of *ACCESS-POINT* is to maximize $\sum_{c=1}^{C} \rho_c \log q_c$, subject to systemwide feasibility constraints on $[q_c]$. Thus, it seeks to achieve weighted proportional fairness among all clients in the system, where the weight of a client equals its payment.

Based on this interpretation, we can design a bidding game that jointly solves the *CLIENT*$_c$ subproblem and the *ACCESS-POINT* subproblem. The bidding game is formulated as follows:

1. Each client c arbitrarily chooses its amount of payment, $\rho_c > 0$, and submits the payment to the AP.
2. The AP gathers payments from all clients, and then chooses a scheduling policy, under which the timely-throughput of clients, $[q_c]$, solves the *ACCESS-POINT* subproblem with parameter $[\rho_c]$.
3. Each client c observes its own timely-throughput, q_c. It computes the parameter $\psi_c = \rho_c / q_c$. It then updates its amount of payment to be the solution of the *CLIENT*$_c$ subproblem with parameter ψ_c, and submits the new payment to the AP.
4. Repeat Step 2.

By Theorem 6, it can be shown that, when the above bidding game converges, it converges to the optimal solution to the *SYSTEM* problem.

Theorem 7. *Let $[\rho_c^*]$, $[\psi_c^*]$, and $[q_c^*]$ be a fixed point of the above bidding game. Then $[q_c^*]$ maximizes $\sum_{c=1}^{C} U_c(q_c)$ over all feasible $[q_c]$ if all three vectors $[\rho_c^*]$, $[\psi_c^*]$, and $[q_c^*]$ are positive.*

8 The Performance of Generalized Transmission Time Policies

Following from the decomposition of the *SYSTEM* problem, we need to find a policy for the AP whose resulting $[q_c]$ solves the *ACCESS-POINT* subproblem. This is apparently a complex problem since it involves 2^C constraints. However, in Section 9, we will show that a very simple on-line scheduling policy for this problem.

The scheduling policy is called the *weighted-transmission time* policy. Let us denote by $f_c(k)$ the number of time slots that the AP spends transmitting the packet for client c in the k^{th} interval. At the beginning of the $(k+1)^{th}$ interval, the weighted-transmission time policy sorts all clients by $\frac{\sum_{j=1}^{k} f_c(j)}{\rho_c}$ and assigns priorities accordingly, with clients with smaller $\frac{\sum_{j=1}^{k} f_c(j)}{\rho_c}$ getting higher priorities.

Before showing that the weighted-transmission time policy solves the *ACCESS-POINT* subproblem, we first determine the precise $[q_c]$ resulting from a general class of policies that includes the weighted-transmission time policy as well as the largest time-debt first policy defined earlier.

Definition 6. *We say that a policy is a* generalized transmission time policy with parameter $[(a_c, b_c)]$*, where $a_c > 0$ and $b_c \geq 0$, if, at the beginning of the $(k+1)^{th}$ interval, the AP sorts all clients according to $a_c \sum_{j=1}^{k} f_c(j) - b_c k$, and assigns higher priorities to clients with lower value of this quantity.*

Note that the weighted-transmission time policy is the special case where $a_c \equiv \frac{1}{\rho_c}$ and $b_c \equiv 0$. Also, the largest time-based debt first policy discussed in Section 5 is also a special case of this class obtained by setting $a_c \equiv 1$ and $b_c \equiv w_c$.

Theorem 8. *For given $[(a_c, b_c)]$, define $\{H_i\}$ and $\{\theta_i\}$ recursively as follows. Let $H_0 := \phi$, $\theta_0 := -\infty$, and*

$$H_i := \arg \min_{S: S \supsetneq H_{i-1}} \frac{(I_{H_{i-1}} - I_S) - \sum_{c \in S \setminus H_{i-1}} b_c/a_c}{\sum_{c \in S \setminus H_{i-1}} 1/a_c}$$

$$\theta_i := \frac{(I_{H_{i-1}} - I_{H_i}) - \sum_{c \in H_i \setminus H_{i-1}} b_c/a_c}{\sum_{c \in H_i \setminus H_{i-1}} 1/a_c}, \text{ for all } i > 0.$$

In selecting H_i, ties are broken arbitrarily. Then, the generalized transmission time policy with parameter $[(a_c, b_c)]$ results in a timely-throughput $q_c = p_c \frac{b_c + \theta_{i(c)}}{a_c}$ for each client c, where $i(c)$ is chosen so that $c \in H_{i(c)} \setminus H_{i(c)-1}$.

Proof. For fixed i, define a subset $Q_i \subseteq \mathbb{R}^C$ by $Q_i := \{[q_c] | q_c \geq p_c \frac{b_c+\theta_i}{a_c}, \forall c \notin H_{i-1}\}$, for all i. The proof consists of two parts. First, we prove that the vector of timely-throughputs under the generalized transmission time policy with parameter $[(a_c, b_c)]$ approaches Q_i, for all i. We then prove that $[q_c] = p_c \frac{b_c+\theta_{i(c)}}{a_c}$ is the only feasible point in $\cap_i Q_i$.

By Lemma 1, the timely-throughput of client c is $\liminf_{K\to\infty} \frac{p_c \sum_{k=1}^{K} f_c(k)}{K}$. Hence,

$$\begin{aligned} q_c \geq p_c \frac{b_c+\theta_i}{a_c} &\Leftrightarrow \liminf_{K\to\infty} \frac{\sum_{k=1}^{K} f_c(k)}{K} \geq \frac{b_c+\theta_i}{a_c} \\ &\Leftrightarrow \liminf_{K\to\infty} \frac{a_c \sum_{k=1}^{K} f_c(k) - b_c K}{\sqrt{a_c} K} \geq \frac{\theta_i}{\sqrt{a_c}}. \end{aligned}$$

Thus, proving that the vector of timely-throughputs approaches Q_i is equivalent to proving that the vector $[\frac{a_c \sum_{k=1}^{K} f_c(k) - b_c K}{\sqrt{a_c} K}]$ approaches $V_i := \{[v_c] | v_c \geq \frac{\theta_i}{\sqrt{a_c}}, \forall c \notin H_{i-1}\}$, as $K \to \infty$.

We apply Theorem 1. Suppose that, at the beginning of the $(K+1)^{th}$ interval, the vector $[\frac{a_c \sum_{k=1}^{K} f_c(k) - b_c K}{\sqrt{a_c} K}]$ is not in V_i. The closest point in V_i to $[\frac{a_c \sum_{k=1}^{K} f_c(k) - b_c K}{\sqrt{a_c} K}]$ is $[d_c(K)] := [v_c | v_c = \frac{a_c \sum_{k=1}^{K} f_c(k) - b_c K}{\sqrt{a_c} K}$, if $c \in H_{i-1}$ or $\frac{a_c \sum_{k=1}^{K} f_c(k) - b_c K}{\sqrt{a_c} K} \geq \frac{\theta_i}{\sqrt{a_c}}; v_c = \frac{\theta_i}{\sqrt{a_c}}$, otherwise$]$. The hyperplane $h([\frac{a_n \sum_{k=1}^{K} f_c(k) - b_c K}{\sqrt{a_c} K}])$ is then $\{x \in \mathbb{R}^C | g(x) = \sum_c (\frac{a_c \sum_{k=1}^{K} f_c(k) - b_c K}{\sqrt{a_c} K} - d_c(K))(x - d_c(K)) = 0\}$. As $g([\frac{a_c \sum_{k=1}^{K} f_c(k) - b_c K}{\sqrt{a_c} K}]) > 0$, we need to show $g([\frac{a_c E\{f_c(K+1)\} - b_c}{\sqrt{a_c}}]) \leq 0$ under the generalized transmission time policy with parameter $[(a_c, b_c)]$.

Without loss of generality, assume that the clients are sorted so that $a_1 \sum_{k=1}^{K} f_1(k) - b_1 K \leq a_2 \sum_{k=1}^{K} f_2(k) - b_2 K \leq \cdots \leq a_m \sum_{k=1}^{K} f_m(k) - b_m K < \theta_i K \leq a_{m+1} \sum_{k=1}^{K} f_{m+1}(k) - b_{m+1} K \leq \ldots$. Then, $g(x) = \sum_{c \notin H_{i-1}, c \leq m} (\frac{a_c \sum_{k=1}^{K} f_c(k) - b_c K}{\sqrt{a_c} K} - \frac{\theta_i}{\sqrt{a_c}})(x - \frac{\theta_i}{\sqrt{a_c}})$. We manipulate the ordering of clients so that clients in H_{i-1} have higher priorities than other clients, while maintaining the relative priorities among clients not in H_{i-1}. This manipulation will not increase the value of $E\{f_c(K+1)\}$ for $c \notin H_{i-1}$. As the value of $g([\frac{E\{f_c(K+1)\} - b_c}{\sqrt{a_c}}])$ is not influenced by $E\{f_c(K+1)\}$ for $c \in H_{i-1}$, and is non-increasing with $E\{f_c(K+1)\}$ for $c \notin H_{i-1}$, this manipulation will not

decrease the value of $g([a_c\frac{E\{f_c(K+1)\}-b_c}{\sqrt{a_c}}])$. Let $c_0 = |H_{i-1}| + 1$. We then have

$$
\begin{aligned}
& g([\frac{E\{f_c(K+1)-b_c\}}{\sqrt{a_c}}]) \\
&= \sum_{c=c_0}^{m}(\frac{a_c\sum_{k=1}^{K} f_c(k) - b_cK}{\sqrt{a_c}K} - \frac{\theta_i}{\sqrt{a_c}})(\frac{a_cE\{f_c(K+1)\}-b_c}{\sqrt{a_c}} - \frac{\theta_i}{\sqrt{a_c}}) \\
&= \sum_{c=c_0}^{m}(\frac{a_c\sum_{k=1}^{K} f_c(k) - b_cK}{K} - \theta_i)(E\{f_c(K+1)\} - \frac{b_c+\theta_i}{a_c}) \\
&= \sum_{c=c_0}^{m-1}(\frac{a_c\sum_{k=1}^{K} f_c(k) - b_cK}{K} - \frac{a_{c+1}\sum_{k=1}^{K} f_{c+1}(k) - b_{c+1}K}{K})[\sum_{l=c_0}^{c}(E\{f_l(K+1)\} - \frac{b_l+\theta_i}{a_l})] \\
&\quad + (\frac{a_m\sum_{k=1}^{K} f_m(k) - b_mK}{K} - \theta_i)[\sum_{l=c_0}^{m}(E\{f_l(K+1)\} - \frac{b_l+\theta_i}{a_l})].
\end{aligned}
$$

Let $\delta_c := (\frac{a_c\sum_{k=1}^{K} f_c(k)-b_cK}{K} - \frac{a_{c+1}\sum_{k=1}^{K} f_{c+1}(k)-b_{c+1}K}{K})$, for $c_0 \le c \le m-1$, and $\delta_m := (\frac{a_m\sum_{k=1}^{K} f_m(k)-b_mK}{K} - \theta_i)$. Then, $\delta_c < 0$, for all $c_0 \le c \le m$. Recall that I_S is the average number of forced idle time slots when the AP only schedules transmissions for clients in S. The above equation can be rewritten as

$$
\begin{aligned}
& g([\frac{E\{f_c(K+1)-b_c\}}{\sqrt{a_c}}]) \\
&= \sum_{c=c_0}^{m}\delta_c[I_{H_{i-1}} - I_{H_{i-1}\cup\{c_0,c_0+1\ldots,m\}} - \sum_{l=c_0}^{c}\frac{b_l+\theta_i}{a_l}] \\
&= \sum_{c=c_0}^{m}\delta_c(\sum_{l=c_0}^{c}\frac{1}{a_l})[\frac{(I_{H_{i-1}} - I_{H_{i-1}\cup\{c_0,c_0+1\ldots,c\}}) - \sum_{l=c_0}^{c} b_l/a_l}{\sum_{l=c_0}^{c} 1/a_l} - \theta_i] \le 0,
\end{aligned}
$$

by the definitions of H_i and θ_i. By Theorem 1, we have established that the vector of timely-throughputs under the generalized transmission time policy with parameter $[(a_c, b_c)]$ approaches Q_i, for all i. It follows that the vector of timely-throughput also approaches $\cap_i Q_i$.

We note that, for each $[q_c] \in \cap_i Q_i$, $q_c \ge p_c\frac{b_c+\theta_{i(c)}}{a_c}$. Further, $\sum_{c=1}^{C}\frac{b_c+\theta_{i(c)}}{a_c} = T - I_{1,2,\ldots,C}$. Since the vector of timely-throughputs $[q_c]$ is feasible only if $\sum_{c=1}^{C}\frac{q_c}{p_c} \le T - I_{1,2,\ldots,C}$, it follows that $[p_c\frac{b_c+\theta_{i(c)}}{a_c}]$ is the only feasible point in $\cap_i Q_i$. This completes the proof. □

9 Optimality of Weighted-Transmission Time Policy for the ACCESS-POINT Problem

Now we turn to the solution of the *ACCESS-POINT* problem. We assume throughout this section that the payment of each client c is fixed at ρ_c. The goal of the AP is to maximize $\sum_{c=1}^{C} \rho_c q_c$, subject to feasibility of the vector $[q_c]$, i.e., $\sum_{c\in S} \frac{q_c}{p_c} + I_S \leq T$ for all S, and $q_c \geq 0$ for all c.

Recall that the weighted-transmission time policy simply prioritizes clients by $\frac{\sum_{j=1}^{k} f_c(j)}{\rho_c}$, and gives higher priority to clients with lower values of $\frac{\sum_{j=1}^{k} f_c(j)}{\rho_c}$.

Utilizing the precise computation of the timely-throughputs that are delivered under generalized transmission time policies, we prove that this weighted-transmission policy solves the *ACCESS-POINT* subproblem.

Theorem 9. *Given positive $[\rho_c]$, the timely-throughput $[q_c^*]$ achieved by applying the weighted-transmission time policy maximizes $\sum_{c=1}^{C} \rho_c q_c$ over all feasible $[q_c]$.*

Proof. Recall that the weighted-transmission policy is a special case of the generalized transmission time policy with parameter $[(\frac{1}{\rho_c}, 0)]$. By Theorem 8, let $H_0 = \phi$, $\theta_0 = -\infty$, and

$$H_i := \arg \min_{S: S \supsetneq H_{i-1}} \frac{I_{H_{i-1}} - I_S}{\sum_{c\in S\setminus H_{i-1}} \rho_c}$$
$$\theta_i := \frac{I_{H_{i-1}} - I_{H_i}}{\sum_{c\in H_i\setminus H_{i-1}} \rho_c}, \text{ for all } i > 0,$$

and we obtain that the timely-throughput of client n is $q_c^* = p_c\rho_c\theta_{i(c)}$.

We first show that $\theta_{i+1} \geq \theta_i > 0$, for all $i > 0$. Now $\theta_1 > 0$ holds since for any $S \neq \phi$, $I_S < T = I_\phi = I_{H_0}$. Further, as I_S is non-increasing with S and $H_{i-1} \subset H_i$, it follows that $\theta_i \geq 0$, for all $i > 1$. We now prove $\theta_{i+1} \geq \theta_i$ by contradiction. Suppose to the contrary that $\theta_{i+1} < \theta_i$. Then, $\sum_{c\in H_{i+1}\setminus H_{i-1}} \rho_c = \sum_{c\in H_{i+1}\setminus H_i} \rho_c + \sum_{c\in H_i\setminus H_{i-1}} \rho_c = (I_{H_i} - I_{H_{i+1}})/\theta_{i+1} + (I_{H_{i-1}} - I_{H_i})/\theta_i > (I_{H_{i-1}} - I_{H_{i+1}})/\theta_i$. Therefore, $\theta_i > \frac{I_{H_{i-1}} - I_{H_{i+1}}}{\sum_{c\in H_{i+1}\setminus H_{i-1}} \rho_c}$, contradicting the construction of H_i and θ_i. Finally, $\theta_i > 0$ since $\theta_i \geq \theta_1$.

Let $\mu_c = 0$, for all c; $\zeta_{H_1} = 1/\theta_1$, $\zeta_{H_i} = 1/\theta_i - 1/\theta_{i-1}$ for $i > 1$, and $\zeta_S = 0$, for all $S \notin \{H_i\}$. We show that the vectors $[q_c^*]$, $[\mu_c]$, and $[\zeta_S]$ satisfy the KKT condition for the *ACCESS-POINT* subproblem (17)–(20). Now (17) holds since

$$-\frac{\rho_c}{q_c^*} + \frac{\sum_{S\ni c} \zeta_S}{p_c} - \mu_c = \frac{-1}{p_c\theta_{i(c)}} + \frac{\sum_{j=1}^{i(c)} (1/\theta_j - 1/\theta_{j-1})}{p_c} = 0.$$

By the construction of $\{H_i\}$ and $\{\theta_i\}$,

$$\sum_{c\in H_i}\frac{q_c^*}{p_c}=\sum_{j=1}^{i}\sum_{c\in H_i\setminus H_{i-1}}\rho_c\theta_i=\sum_{j=1}^{i}(I_{H_{j-1}}-I_{H_j})=T-I_{H_i},$$

for all i. Hence, $\zeta_S[\sum_{c\in S}\frac{q_c^*}{p_c}-(T-I_S)]=0$, for all S, and (18) holds. Finally, (19) and (20) are also satisfied since $\mu_c q_c^*=0\times q_c^*=0$ and $\zeta_S\geq 0$. Thus, $[q_c^*]$ solves *ACCESS-POINT* with parameter $[\rho_c]$ and maximizes $\sum_{c=1}^{C}\rho_c\log q_c$ over all feasible $[q_c]$. □

10 Simulation Results

We now present simulation results concerning the three policies developed in this paper, admission control and packet scheduling for inelastic traffic, and utility maximization for elastic traffic. Our simulations are based on the G.711 codec for VoIP, and the IEEE 802.11b standard, conducted under the ns-2 simulation environment. Table 1 summarizes the relevant parameters of our simulations. All results presented in this section are averages of 20 runs of simulations.

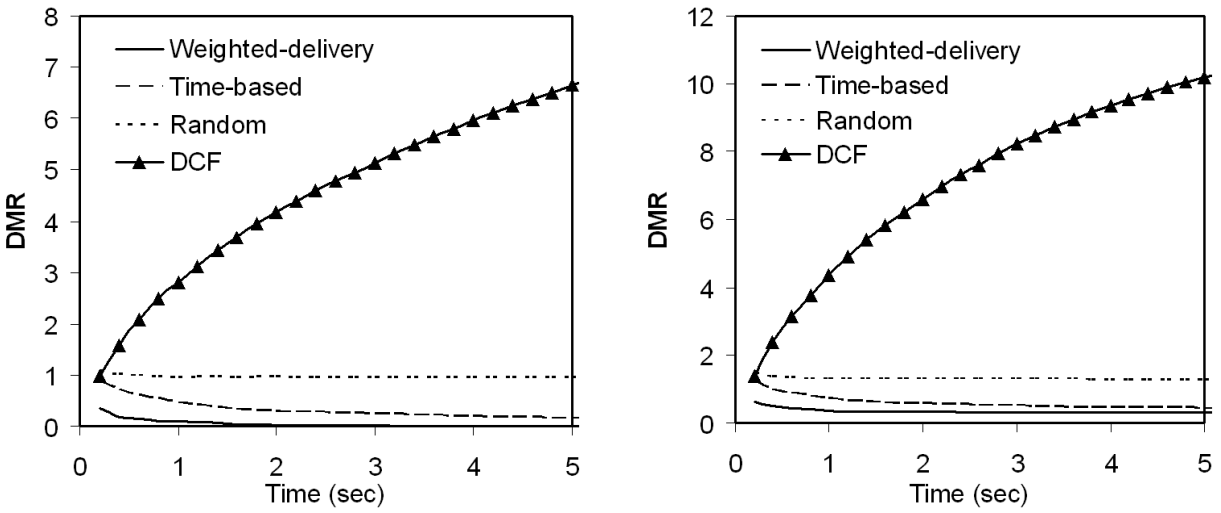
Table 1: Simulation Setup

Packetization interval	20 ms
Payload size per packet	160 Bytes
Transmission data rate	11 Mb/s
Transmission time (including MAC overheads)	610 μs
# of time slots in an interval	32

10.1 Inelastic Traffic

We first consider scenarios where clients carry inelastic traffic; that is, each client has a specified timely-throughput requirement. We consider two groups of clients, group A and group B. Each client in group A requires a timely-throughput of 0.99 packets per interval. On the other hand, each client in group B only requires a timely-throughput of 0.8 packets per interval. We assume that the channel reliability, p_c, of the c^{th} client in each group is $(60+c)\%$. Using our admission control algorithm, we find that a system with 11 group A clients and 12 group B clients is feasible, while a system with 12 group A clients and 12 group B clients is not.

We compare our two largest debt first scheduling policies against two other policies. One of the policies is the default mechanism of the 802.11 DCF, and the other is a random policy



(a) Performance of a feasible set

(b) Performance of an infeasible set

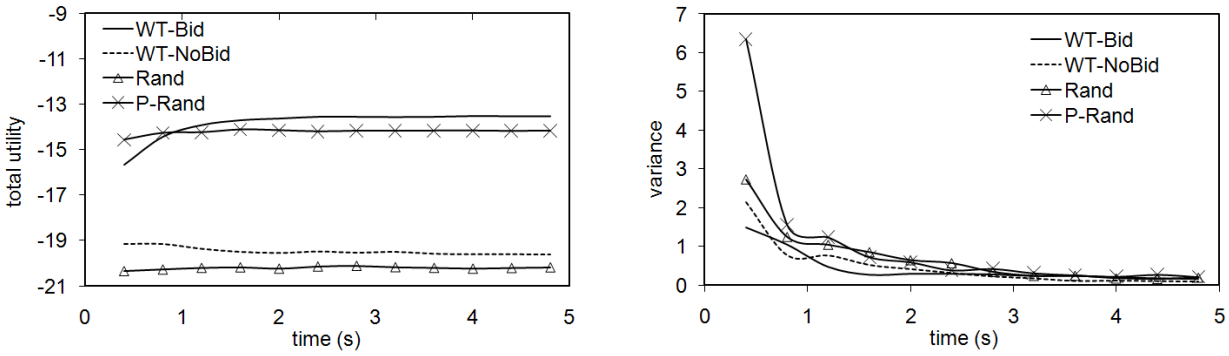
Figure 3: Total delivery debt for inelastic traffic

where the AP assigns priorities randomly to all clients at the beginning of each interval. We use the total delivery debt in the system, $\sum_c (q_c - \text{actual timely-throughput of client } c)^+$, as the performance metric. Note that a system is fulfilled if and only if the total delivery debt converges to zero.

The simulation results are shown in Figure 3. For the feasible system, the total delivery debts under the two largest debt first policies converge to zero quickly, showing that they both fulfill the system. On the other hand, both DCF and the random policy have much larger total delivery debts. This suggests that these current standard mechanisms are suboptimal when handling real-time flows. In Figure 3b, we also show that the total delivery debts of all the four policies remain bounded away from zero for the infeasible system. This result confirms that our admission control policy is indeed accurate. Though infeasible, among all the four policies, the two largest debt first scheduling policies still achieve the minimum total delivery debt. Thus, our proposed policies can still provide better performance even when the requirements of clients are infeasible.

10.2 Elastic Traffic

We consider a system with 30 wireless clients. The utility function of client c is set to be $U_c(q_c) = \gamma_c \frac{q_c^{\alpha_c}-1}{\alpha_c}$, where $\gamma_c > 0$ and $0 < \alpha_c < 1$. By tuning the parameters γ_c and α_c, we can generate a



(a) Average of total utility

(b) Variance of total utility

Figure 4: Performance of the first setting

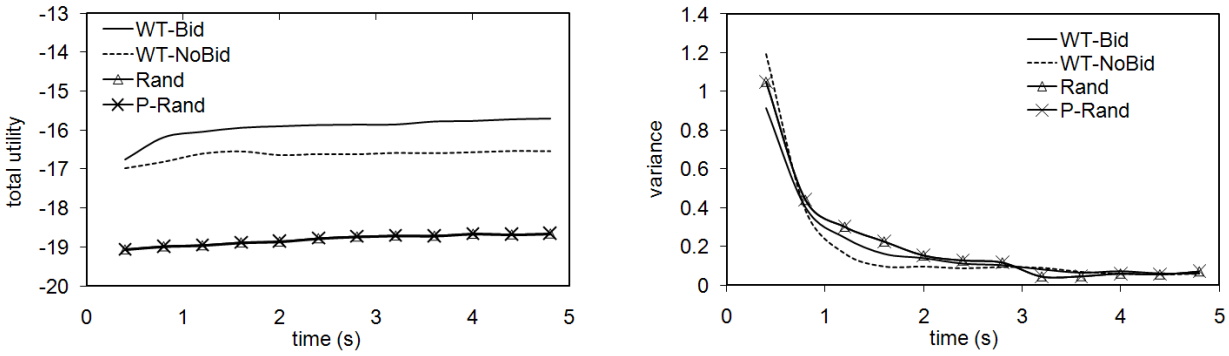
family of utility functions.

We compare four different policies. The first is our proposed policy, which consists of applying the weighted-transmission time policy as well as the bidding game (WT-Bid). The second is one that only uses the weighted-transmission time policy but does not employ the bidding game (WT-NoBid); that is, clients do not update their payments ρ_c. We then consider a random policy (Rand) where the AP randomly assigns priorities to clients at the beginning of each interval. Finally, we also evaluate a policy where the AP gives higher priorities to clients with larger γ_c, and break ties randomly (P-Rand). We evaluate the performance of each policy through its average and variance of total utility over the 20 simulation runs.

We consider two settings. In the first setting, we choose $p_c = (50+c)\%$, $\gamma_c = (c \mod 3)+1$, and $\alpha_c = 0.3 + 0.1(c \mod 5)$. In the second setting, we set $p_c = (20 + 2c)\%$, $\gamma_c = 1$, and $\alpha_c = 0.3 + 0.1(c \mod 5)$. Simulation results for the two settings are shown in Figure 4 and Figure 5, respectively. In both settings, our proposed policy achieves the highest total utility in the system among all the four policies. In addition, it also converges very quickly, as its variances are small for both settings.

11 Concluding Remarks

We have presented an account of a theory for serving real-time flows in wireless systems. We have proposed a model that jointly captures the delay bounds and timely-throughput requirements of clients, as well as the unreliable nature of wireless transmissions. We have derived a necessary and sufficient condition for a system to be feasible. Based on the condition, we have



(a) Average of total utility

(b) Variance of total utility

Figure 5: Performance of the second setting

proposed an efficient algorithm for precise admission control. Two on-line scheduling policies have also been proposed and proved to be feasibility optimal.

Further, we have considered the scenario where the timely-throughput requirements of the real-time flows are elastic. We have formulated this as a utility maximization problem. It can be further decomposed into two subproblems, which can be implemented as a bidding game among clients and the AP. In this bidding game, clients apply a simple selfish strategy. We have discussed the strategy for the AP in the bidding game and presented a simple on-line scheduling policy for solving the problem faced by the access-point in this decomposition.

References

[1] I.-H. Hou, V. Borkar, and P. Kumar, "A Theory of QoS for Wireless," in *Proc. of INFOCOM*, 2009.

[2] I.-H. Hou and P. Kumar, "Utility Maximization for Delay Constrained QoS in Wireless," in *Proc. of INFOCOM*, 2010.

[3] I.-H. Hou and P. Kumar, "Admission Control and Scheduling for QoS Guarantees for Variable-Bit-Rate Applications on Wireless Channels," in *Proc. of MobiHoc*, 2009.

[4] I.-H. Hou and P. Kumar, "Scheduling Heterogeneous Real-Time Traffic over Fading Wireless Channels," in *Proc. of INFOCOM*, 2010.

[5] I.-H. Hou and P. Kumar, "Utility-Optimal Scheduling in Time-Varying Wireless Networks with Delay Constraints," in *Proc. of MOBIHOC*, 2010.

[6] I.-H. Hou and P. Kumar, "Broadcasting Delay-Constrained Traffic over Unreliable Wireless Links with Network Coding," in *Proc. of MOBIHOC*, 2011.

[7] F. Kelly, "Charging and rate control for elastic traffic," *European Trans. on Telecommunications*, vol. 8, pp. 33–37, 1997.

[8] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, pp. 237–252, 1998.

[9] E. Eisenberg and D. Gale, "Consensus of subjective probabilities: The pari-mutuel method," *The Annals of Mathematical Statistics*, vol. 30, pp. 165–168, Mar. 1959.

[10] Y. Xiao and H. Li, "Evaluation of distributed admission control for the IEEE 802.11e EDCA," *Communications Magazine, IEEE*, vol. 42, Sept. 2004.

[11] D. Pong and T. Moors, "Call admission control for IEEE 802.11 contention access mechanism," in *Proc. of GLOBECOM 2003*.

[12] S. Garg and M. Kappes, "Admission control for VoIP traffic in IEEE 802.11 networks," in *Proc. of GLOBECOM 2003*.

[13] H. Zhai, X. Chen, and Y. Fang, "A call admission and rate control scheme for multimedia support over IEEE 802.11 wireless LANs," *Wireless Networks*, vol. 12, August 2006.

[14] S. Shin and H. Schulzrinne, "Call admission control in IEEE 802.11 WLANs using QP-CAT," in *Proc. of INFOCOM 2008*.

[15] D. Gao, J. Cai, and K. Ngan, "Admission control in IEEE 802.11e wireless LANs," *IEEE Network*, pp. 6–13, July/August 2005.

[16] D. Niyato and E. Hossain, "Call admission control for QoS provisioning in 4G wireless networks: issues and approaches," *IEEE Network*, pp. 5–11, September/October 2005.

[17] M. Ahmed, "Call admission control inwireless networks: A comprehensive survey," *IEEE Communications Surveys*, vol. 7, no. 1, pp. 50–69, 2005.

[18] L. Tassiulas and A. Ephremides, "Dynamic server allocation to parallel queues with randomly varying connectivity," *IEEE Trans. on Information Theory*, vol. 39, March 1993.

[19] M. J. Neely, "Delay analysis for max weight opportunistic scheduling in wireless systems," in *Proc. of Allerton Conf.*, 2008.

[20] K. B. Johnsson and D. C. Cox, "An adaptive cross-layer scheduler for improved QoS support of multiclass data services on wireless systems," *IEEE J. on Selected Areas in Communications*, vol. 23, no. 2, 2005.

[21] A. Dua and N. Bambos, "Deadline constrained packet scheduling for wireless networks," in *62nd IEEE VTC Fall 2005*.

[22] V. Raghunathan, V. Borkar, M. Cao, and P. Kumar, "Index policies for real-time multicast scheduling for wireless broadcast systems," in *Proc. of IEEE INFOCOM 2008*.

[23] S. Shakkottai and R. Srikant, "Scheduling real-time traffic with deadlines over a wireless channel," *Wireless Networks*, vol. 8, Jan. 2002.

[24] H. Fattah and C. Leung, "An overview of scheduling algorithms in wireless multimedia networks," *IEEE Wireless Communications*, vol. 9, pp. 76–83, Oct. 2002.

[25] Y. Cao and V. Li, "Scheduling algorithms in broadband wireless networks," *Proceedings of the IEEE*, vol. 89, pp. 76–87, Jan. 2001.

[26] X. Lin and N. Shroff, "Utility maximization for communication networks with multipath routing," *IEEE Trans. on Automated Control*, vol. 51, no. 5, pp. 766–781, 2006.

[27] M. Xiao, N. Shroff, and E. Chong, "A utility-based power-control scheme in wireless cellular systems," *IEEE/ACM Trans. on Networking*, vol. 11, no. 2, pp. 210–221, 2003.

[28] Y. Cao and V. Li, "Utility-oriented adaptive QoS and bandwidth allocation in wireless networks," in *Proc. of ICC*, 2002.

[29] G. Bianchi, A. Campbell, and R. Liao, "On utility-fair adaptive services in wireless networks," in *Proc. of IWQoS*, pp. 256–267, 1998.

[30] M. Loeve, *Probability Theory II*. Springer-Verlag, 1978.

[31] D. Blackwell, "An analog of the minimax theorem for vector payoffs," *Pacific J. Math*, vol. 6, no. 1, 1956.

[32] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*. MIT Press, 2009.