Optimal Scheduling of Real-Time Messages in Peer-to-Peer Wireless Networks

Juan José Jaramillo, Shihuan Liu and Lei Ying
Dept. of Electrical and Computer Engineering
Iowa State University
Email: {jjjarami, liush08, leiying}@iastate.edu

***Abstract*—This paper studies the problem of service allocation and scheduling of real-time messages in peer-to-peer wireless networks. Using stochastic network theory and optimization we propose a model that allows us to design a dynamic service allocation algorithm that maximizes the total network utility while meeting deadline constraints, by appealing to connections between Lagrange multipliers and deficits in service. The model allows for general interference constraints and arrival models. Using simulations we compare our algorithm against an optimal solution proposed for scheduling persistent real-time traffic and show the limitations of that approach to handle real-time messages for providing fairness.**

I. Introduction

With the ever increasing popularity of wireless networks brought by their flexibility and resilience, new and exciting applications have started to emerge. No longer wireless networks are expected to just transmit persistent data flows and real-time traffic such as voice calls and streaming video, but they are also expected to be used for transmitting small-sized files such as text messages and pictures. It has been shown in [1] that scheduling algorithms such as the MaxWeight scheduler [2], [3] that provide maximum throughput for persistent data flows do not perform well with small-sized files. Novel solutions have been developed in [1], [4]–[6] to schedule small-sized files *without* delay constraints.

In this paper, we study the newly developed FlashLinQ [7] peer-to-peer communication technology over licensed spectrum. The technology allows devices to discover each other within a geographical region, enabling single-hop device-to-device communications for a fixed time interval called a *frame*. This allows *location-aware* communications in *real-time*. We model it as a peer-to-peer wireless network where multiple transmitter-receiver pairs request communication at the beginning of the frame, where a frame is a contiguous set of time-slots of fixed duration. Each transmitter intends to send one fixed-size message to its receiver. Messages are either delivered within the frame or dropped, i.e., all messages are associated with a hard delay constraint equal to one frame, and are *real-time messages*. We assume the transmissions are scheduled by a central omniscient authority that can do the optimal scheduling since it knows the location of the wireless devices, and allow us to determine the best this type of communications can achieve.

Research supported by NSF Grant 09-53165, DTRA grants HDTRA1-08-1-0016 and HDTRA1-09-1-0055.

In this paper we thus address the problem of service allocation and scheduling of real-time message requests under strict delay constraints, and present an algorithm that optimally allocates service to users while meeting deadline constraints.

The main contributions of this work are summarized as follows:

1) We propose an optimization formulation for the problem of service allocation and scheduling of real-time messages under strict per-message deadline constraints in location-aware, wireless peer-to-peer networks. We show that using the fact that the network is aware of the location of the devices allows us to deal with the difficulty of scheduling small-sized messages, translating the problem of serving message requests into a long-term formulation where messages are grouped by regions where channel and interference conditions are similar. The formulation allows for very general interference constraints and arrival models.
2) We design an optimal service controller and scheduler that allocates service such that it maximizes the total network utility in a stochastic sense, while meeting deadline constraints.
3) Using simulations we compare our algorithm against an optimal solution proposed for scheduling persistent real-time traffic and show the limitations of that approach to handle real-time messages for providing fairness, and the need to develop a new approach.

II. Network Model

In this section we describe the model we propose for a network that has message requests subject to deadline constraints. The network is located in a bounded region $\mathcal{R}$, where at the beginning of each frame, multiple communication requests occur in the network. A communication request is from one location of the region to another location. The request either gets fulfilled during that frame, or gets dropped due to deadline expiration. In this network, all flows are finite-sized messages with strict deadlines, so resource allocation algorithms designed for persistent flows cannot be used. To effectively schedule these real-time messages, we propose to partition $\mathcal{R}$ into subregions that share similar interference and

channel conditions. This will allow us to pose the problem as a long-term optimization problem, where we can maximize the total network utility.

Traffic requests are assumed to originate in a region $\mathcal{R}$ that we divide in M disjoint subregions $\{r_i\}_{i\in\mathcal{M}}$, i.e., $r_i \cap r_j = \emptyset$ for all $i \neq j \in \mathcal{M} \stackrel{def}{=} \{1,\ldots,M\}$ and $\cup_{i\in\mathcal{M}} r_i = \mathcal{R}$. Thus, to specify a flow, we must specify the region where the source node is and the region where the destination is. These regions are also used to define the interference constraints, which we represent by the interference graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is the set of vertices and $\mathcal{E}$ is the set of edges. Formally, $\mathcal{V} \stackrel{def}{=} \{v = (r_i, r_j) : r_i, r_j \in \mathcal{R} \text{ for } i,j \in \mathcal{M}\}$ denotes any pair of regions such that the source node is in r_i and the destination is in r_j, and if $(v_1, v_2) \in \mathcal{E}$, where $v_1 = (r_{i1}, r_{j1})$, $v_2 = (r_{i2}, r_{j2})$, then a flow with source in r_{i1} and destination in r_{j1} cannot be scheduled to transmit simultaneously with a flow with source in r_{i2} and destination in r_{j2}.

We assume that time is divided in *slots*, and a set of T consecutive time slots is called a *frame*. Every message is assumed to be comprised of a single, fixed-size packet such that the packet can be transmitted in a time slot and has a deadline of T slots. Furthermore, it is assumed that all packets arrive at the beginning of the frame.

Let $a = (a_{ij})_{i,j\in\mathcal{M}}$ denote the number of real-time messages that arrive at region r_i destined for region r_j at the beginning of a given frame. We assume that a_{ij} is a random variable with mean λ_{ij} and variance σ_{ij}^2, that is independent between different frames, and is such that $Pr(a_{ij} = 0) > 0$ and $Pr(a_{ij} = 1) > 0$. The last assumption is to guarantee that the Markov chain we define later is both irreducible and aperiodic, but it can be substituted by other similar assumptions.

Depending on the wireless technology used, we can have some channel feedback before or after a transmission occurs, either in the form of channel estimation or receiver feedback, respectively. Furthermore, we can define different channel models depending on whether the receivers acknowledge reception of all the packets at the end of the frame, or if acknowledgments are received after each transmission.

In this paper we assume that the potential number of packets that can be transmitted from region r_i destined for region r_j in a given time slot is denoted by $c = (c_{ij})_{i,j\in\mathcal{M}}$. We assume that the channel state c_{ij} is a Bernoulli random variable, is known at the beginning of the frame, and remains constant for the entire frame. Furthermore, we assume that the channel state is independent between different frames and independent of arrivals. This channel model corresponds to the case when we do channel estimation before transmissions occur. We use this model because it allow us to explain the main ideas behind our algorithm in the simplest way. The other cases have a development similar in nature and are thus omitted. The interested reader is referred to [8] for a related problem where these different channel models are studied in further detail, and it is shown how different channel models affect scheduling.

Denote by $q = (q_{ij})_{i,j\in\mathcal{M}}$ the minimum fraction of packets that need to be served originating in region r_i destined for region r_j, and by $U_{ij}(q_{ij})$ the utility function associated with such fraction. Furthermore, we assume that the function $U_{ij}()$ is concave.

We denote by $s = (s_{ijt})_{i,j\in\mathcal{M},t\in\mathcal{T}}$ the schedule at any given frame, where s_{ijt} indicates the number of packets scheduled for service from region r_i to region r_j in time slot $t \in \mathcal{T} \stackrel{def}{=} \{1,\ldots,T\}$. Furthermore, we consider only schedules that fulfill all interference constraints, that is, if $s_{i_1j_1t} > 0$ and $s_{i_2j_2t} > 0$ for any given time slot t, then it must be the case that $(v_1, v_2) \notin \mathcal{E}$, where $v_1 = (r_{i1}, r_{j1})$, $v_2 = (r_{i2}, r_{j2})$. Since the number of available messages and the channel state determine the maximum number of packets that can be scheduled, we have the following constraints in the schedule:

$$\sum_{t\in\mathcal{T}} s_{ijt} \leq a_{ij} \text{ for all } i,j \in \mathcal{M} \quad (1)$$

$$s_{ijt} \leq c_{ij} \text{ for all } i,j \in \mathcal{M},\ t \in \mathcal{T} \quad (2)$$

We will denote by $\mathcal{S}(a,c)$ the set of feasible schedules for fixed arrivals and channel state, subject to (1), (2) and the interference constraints given by graph $\mathcal{G}$.

III. Optimization Formulation

We now present a static optimization problem which will be the base to design a dynamic algorithm using a dual decomposition approach.

Our goal is to find a scheduling policy $Pr(s|a,c)$, which is the probability of using schedule $s \in \mathcal{S}(a,c)$ when the arrivals are a and the channel state is c. Thus, the expected service for requests with source in region r_i and destination in r_j, $\mu_{ij}(a,c)$, has the following constraint

$$\mu_{ij}(a,c) \leq \sum_{s\in\mathcal{S}(a,c)} \sum_{t\in\mathcal{T}} s_{ijt} Pr(s|a,c),$$

and the overall expected service is given by

$$\mu_{ij} = \sum_{a,c} \mu_{ij}(a,c) Pr(a) Pr(c).$$

For notational simplicity, define the capacity region for fixed arrivals and channel state as

$$\mathcal{C}(a,c) \stackrel{def}{=} \left\{ \begin{array}{l} (\bar{\mu}_{ij})_{i,j\in\mathcal{M}} : \text{there exists } \bar{s} \in \mathcal{S}(a,c)_{\mathcal{CH}}, \\ \bar{\mu}_{ij} \leq \sum_{t\in\mathcal{T}} \bar{s}_{ijt} \text{ for all } i,j \in \mathcal{M} \end{array} \right\},$$

where $\mathcal{S}(a,c)_{\mathcal{CH}}$ is the convex hull of $\mathcal{S}(a,c)$. Similarly, if we define the overall capacity of the network as $\mathcal{C} \stackrel{def}{=}$

$$\left\{ \begin{array}{l} (\mu_{ij})_{i,j\in\mathcal{M}} : \text{there exists } \bar{\mu}(a,c) \in \mathcal{C}(a,c) \\ \text{for all } a,c \text{ and } \mu_{ij} = E[\bar{\mu}_{ij}(a,c)] \text{ for all } i,j \in \mathcal{M} \end{array} \right\}$$

then we have that $\mu \stackrel{def}{=} (\mu_{ij})_{i,j\in\mathcal{M}} \in \mathcal{C}$.

From the definition of q_{ij} we have the constraint

$$\lambda_{ij} q_{ij} \leq \mu_{ij} \text{ for all } i,j \in \mathcal{M}.$$

In other words, the service rate has to be larger than the minimum number of packets that need to be served.

Since each traffic flow is a real-time message, we cannot define a long-term utility for a flow. However, we can define utility functions for subregions. The goal is therefore to allocate the communication resource fairly to subregions instead of users. The type of fairness is defined by the selected utility function. So, the problem is formulated as follows:

$$\max_{\mu\in\mathcal{C}, 0\le q_{ij}\le 1} \sum_{i,j\in\mathcal{M}} U_{ij}(q_{ij}) \tag{3}$$

subject to

$$\lambda_{ij} q_{ij} \le \mu_{ij} \text{ for all } i,j \in \mathcal{M}.$$

We will denote the optimal solution by μ^*, q^*.

IV. A Duality Theory Approach

Using duality theory, we will show how to solve the optimization problem by solving a set of related subproblems. This problem decomposition will be the basis for the online algorithm that we will present in the next section.

The associated dual function [9] for (3) is

$$D(\delta) \stackrel{def}{=} \max_{\mu\in\mathcal{C}, 0\le q_{ij}\le 1} \sum_{i,j\in\mathcal{M}} U_{ij}(q_{ij}) - \delta_{ij}(\lambda_{ij} q_{ij} - \mu_{ij}).$$

Since the utility function is concave, and the constraints are affine functions, Slater's condition [10] implies that the duality gap is zero and therefore $D(\delta^*) = \sum_{i,j\in\mathcal{M}} U_{ij}(q^*_{ij})$, where

$$\delta^* \in \arg\min_{\delta_{ij}\ge 0} D(\delta)$$

and q^* is the solution to (3). Furthermore, to solve the optimization problem using the dual function, we note that we can simply solve the following subproblems

$$\max_{0\le q_{ij}\le 1} U_{ij}(q_{ij}) - \delta_{ij}\lambda_{ij} q_{ij}$$

and

$$\max_{\mu\in\mathcal{C}} \sum_{i,j\in\mathcal{M}} \delta_{ij}\mu_{ij}. \tag{4}$$

Since $\delta_{ij} \ge 0$ for all $i,j \in \mathcal{M}$, the optimization in (4) has a linear objective, and the service rate is a convex combination of the feasible schedules, we can further decompose (4) as follows

$$\max_{s\in\mathcal{S}(a,c)} \sum_{i,j\in\mathcal{M}} \delta_{ij} \sum_{t\in\mathcal{T}} s_{ijt}.$$

We can then use the following iterative algorithm to find the solution to our optimization problem, where k is the step index:

$$\tilde{q}^*_{ij}(k) \in \arg\max_{0\le q_{ij}\le 1} U_{ij}(q_{ij}) - \delta_{ij}(k)\lambda_{ij} q_{ij}$$

and

$$\tilde{s}^*(a,c,k) \in \arg\max_{s\in\mathcal{S}(a,c)} \sum_{i,j\in\mathcal{M}} \delta_{ij}(k) \sum_{t\in\mathcal{T}} s_{ijt},$$

with update equation

$$\delta_{ij}(k+1) = \{\delta_{ij}(k) + \epsilon[\lambda_{ij}\tilde{q}^*_{ij}(k) - \tilde{\mu}^*_{ij}(k)]\}^+,$$

step-size parameter $\epsilon > 0$ and

$$\tilde{\mu}^*_{ij}(k) \stackrel{def}{=} \sum_{a,c}\sum_{t\in\mathcal{T}} \tilde{s}^*_{ijt}(a,c,k) Pr(a) Pr(c).$$

Using the change of variables $\epsilon\hat{d}_{ij}(k) = \delta_{ij}(k)$ we can rewrite the problem as

$$\tilde{q}^*_{ij}(k) \in \arg\max_{0\le q_{ij}\le 1} \frac{1}{\epsilon} U_{ij}(q_{ij}) - \hat{d}_{ij}(k)\lambda_{ij} q_{ij}$$

and

$$\tilde{s}^*(a,c,k) \in \arg\max_{s\in\mathcal{S}(a,c)} \sum_{i,j\in\mathcal{M}} \hat{d}_{ij}(k) \sum_{t\in\mathcal{T}} s_{ijt},$$

with update equation

$$\hat{d}_{ij}(k+1) = [\hat{d}_{ij}(k) + \lambda_{ij}\tilde{q}^*_{ij}(k) - \tilde{\mu}^*_{ij}(k)]^+.$$

It must be noted that given the update equation of $\tilde{q}^*_{ij}(k)$, we can give a controller interpretation to it, where we regulate the minimum service we give to traffic requests. Similarly, $\hat{d}_{ij}(k)$ can have a queue interpretation, with the the number of arrivals given by $\lambda_{ij}\tilde{q}^*_{ij}(k)$ and the departures given by $\tilde{\mu}^*_{ij}(k)$.

V. Online Algorithm

In this section we first present our online algorithm and subsequently we present its performance analysis.

A. Scheduler and Service Controller

We propose to use the following scheduler when the arrivals and channel state at frame k are given by $a(k)$ and $c(k)$, respectively:

$$\tilde{s}^*(a(k),c(k),d(k)) \in \arg\max_{s\in\mathcal{S}(a(k),c(k))} \sum_{i,j\in\mathcal{M}} d_{ij}(k) \sum_{t\in\mathcal{T}} s_{ijt}, \tag{5}$$

and the following service controller

$$\tilde{q}^*_{ij}(a(k),d(k)) \in \arg\max_{0\le q_{ij}\le 1} \frac{1}{\epsilon} U_{ij}(q_{ij}) - d_{ij}(k) a_{ij}(k) q_{ij}. \tag{6}$$

In the notation we make explicit the fact that the scheduler and the service controller are a function of the arrivals, the channel state, and the parameter $d(k)$.

We need to translate the minimum service to message requests, which is a fraction, into a minimum number of packets that need to be served. This conversion can be made in different ways: we assume that the minimum number of packets to be served at region r_i destined to region r_j, $\tilde{a}_{ij}(k)$, are a binomial random variable with parameters $a_{ij}(k)$ and $\tilde{q}^*_{ij}(a(k),d(k))$. The quantity $\tilde{a}_{ij}(k)$ can be generated by the network as follows: for every message request, flip a coin with probability of *heads* equal to $\tilde{q}^*_{ij}(a(k),d(k))$, and let $\tilde{a}_{ij}(k)$ be the number of *heads* that we get.

The update equation for $d(k)$ is given by

$$d_{ij}(k+1) = [d_{ij}(k) + \tilde{a}_{ij}(k) - \tilde{I}^*_{ij}(a(k),c(k),d(k))]^+,$$

where

$$\tilde{I}^*_{ij}(a(k),c(k),d(k)) \stackrel{def}{=} \sum_{t\in\mathcal{T}} \tilde{s}^*_{ijt}(a(k),c(k),d(k)).$$

We interpret the parameter $d_{ij}(k)$ as a virtual queue that keeps track of the deficit in service for traffic requests from region r_i to region r_j, given the minimum service allocated by our controller.

B. Performance Analysis

We will first bound the expected drift of the Markov chain $d(k)$ for a suitable Lyapunov function. For the sake of readability, we will defer the proofs to the appendixes.

Lemma 1: Consider the Lyapunov function $V(d) = \frac{1}{2}\sum_{i,j\in\mathcal{M}} d_{ij}^2$. If there exists $\tilde{\mu} \in \mathcal{C}$ and $0 \le \tilde{q}_{ij} \le 1$ for all $i,j \in \mathcal{M}$ such that

$$\lambda_{ij}\tilde{q}_{ij} < \tilde{\mu}_{ij} \text{ for all } i,j \in \mathcal{M} \tag{7}$$

then

$$E\left[V(d(k+1))|d(k)=d\right] - V(d) \le B_1 - B_2 \sum_{i,j\in\mathcal{M}} d_{ij} - \frac{1}{\epsilon}\sum_{i,j\in\mathcal{M}} \left\{U_{ij}(\tilde{q}_{ij}) - E\left[U_{ij}(\tilde{q}^*_{ij}(a(k),d))\right]\right\}$$

for some positive constants B_1, B_2, any $\epsilon > 0$, where $\tilde{q}^*(a(k),d)$ is the solution to (6). ◇

Since $d(k)$ defines an irreducible and aperiodic Markov chain, and the last term of the right hand side of the inequality can be bounded, Lemma 1 implies that $d(k)$ is positive recurrent since the expected drift is negative but for a finite set of values of $d(k)$. Thus, a direct consequence of Lemma 1 is the fact that the total service deficit has an $O(\frac{1}{\epsilon})$ bound.

Corollary 1: If there exists $\tilde{\mu} \in \mathcal{C}$, $0 \le \tilde{q}_{ij} \le 1$ for all $i,j \in \mathcal{M}$ such that (7) is true, then the total expected service deficit is upper-bounded by

$$\limsup_{k\to\infty} E\left[\sum_{i,j\in\mathcal{M}} d_{ij}(k)\right] \le B_3 + \frac{1}{\epsilon}B_4,$$

where $B_3 = B_1/B_2$ and

$$B_4 \le \frac{\sum_{i,j\in\mathcal{M}} \max_{0\le q_{ij}\le 1} 2|U_{ij}(q_{ij})|}{B_2}.$$

◇

Before we can prove that our algorithm can achieve the optimal value of (3) in some stochastic sense, we need a related result to Lemma 1.

Lemma 2: Consider the Lyapunov function $V(d) = \frac{1}{2}\sum_{i,j\in\mathcal{M}} d_{ij}^2$. Then

$$E\left[V(d(k+1))|d(k)=d\right] - V(d) \le B_1 - B_2 \sum_{i,j\in\mathcal{M}} d_{ij} - \frac{1}{\epsilon}\sum_{i,j\in\mathcal{M}} \left\{U_{ij}(q^*_{ij}) - E\left[U_{ij}(\tilde{q}^*_{ij}(a(k),d))\right]\right\}$$

for $B_1 > 0$, some nonnegative constant B_2, any $\epsilon > 0$, where q^* is the solution to (3) and $\tilde{q}^*(a(k),d)$ is the solution to (6). Furthermore, if $\lambda_{ij}q^*_{ij} < \mu^*_{ij}$ for all $i,j \in \mathcal{M}$ then $B_2 > 0$. ◇

The difference between Lemma 1 and Lemma 2 is that Lemma 2 does not guarantee that the Markov chain is positive recurrent, but it allows us to compare the expected drift to the optimal solution. The proof technique is identical to the proof for Lemma 1, so it is omitted. With this result, we can now prove that our algorithm is within $O(\epsilon)$ of the optimal value.

Theorem 1: For any $\epsilon > 0$ we have that

$$\limsup_{K\to\infty} \sum_{i,j\in\mathcal{M}} \left\{U_{ij}(q^*_{ij}) - U_{ij}\left(E\left[\frac{1}{K}\sum_{k=1}^{K} \tilde{q}^*_{ij}(a(k),d(k))\right]\right)\right\} \le B\epsilon$$

for some $B > 0$, where q^* is the solution to (3) and $\tilde{q}^*(a(k),d(k))$ is the solution to (6). ◇

From Corollary 1 and Theorem 1 we observe that there is a tradeoff when choosing ϵ, since the more we approach the optimal solution, the larger the total deficit counters will be.

The statement and proofs of Lemma 1 and Theorem 1 follow the techniques in [11], which are similar to the techniques in [12]. Slightly different results can be derived using the techniques in [13] and [14].

VI. Simulations

We now compare the algorithm introduced in Section V-A against the scheduler proposed in [11] for handling persistent real-time traffic. Since the algorithm is an extension to the MaxWeight scheduler for real-time traffic, in this paper we will call it the *real-time MaxWeight algorithm*, while we will call our algorithm the *fraction-based algorithm*.

1) Simulation settings: We divide the region where traffic is generated in two subregions. In other words, $\mathcal{M} = \{1,2\}$. There are 80 users in the network, where 40 users are located in subregion 1, and 40 users are in subregion 2. Each frame consists of 4 time slots, i.e., $T = 4$. We assume the channel between any two regions is always on, in other words, $c_{ij} = 1$ for all $i,j \in \mathcal{M}$. At every frame each user generates a message with probability P_m, where the destination is randomly selected. We will denote by (i,j) the set of transmission requests with source in region r_i and destination in r_j, for $i,j \in \mathcal{M}$. Thus, the aggregate number of message requests in (i,j) will determine a_{ij}. The interference graph for our simulations is given by Fig. 1, where each vertex represents any pair of regions, and an edge joins them if they cannot simultaneously transmit. For example, only transmissions $(1,1)$ and $(2,2)$ can be simultaneously scheduled without interfering with each other. In the simulations, we assume the utility function is α-fair, i.e., $U_{ij}(q_{ij}) = \frac{q_{ij}^{1-\alpha}}{1-\alpha}$. For the fraction-based algorithm, we set $\epsilon = 0.1$ and $\alpha \to 1$, which corresponds to the limit case of proportional fairness.

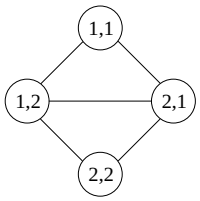

Fig. 1. Interference Graph Used in the Simulations

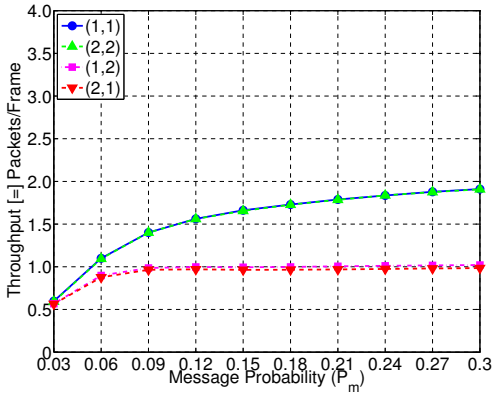

Fig. 2. Throughput for the Fraction-based Algorithm

2) Results: To compare both algorithms, we measure the throughput, which is defined to be the number of packets that are successfully transmitted per frame for every region pair (i, j). In Figs. 2 and 3 we observe that while the fraction-based algorithm tries to fairly allocate the throughput among different region pairs, the real-time MaxWeight algorithm disproportionally gives preference to intraregion transmissions, starving cross-region transmissions.

To understand this behavior, note that the real-time MaxWeight algorithm uses as weights the deficit in service for every flow. Since each traffic request consists of a single packet, then no flow has a deficit that allows it to gain priority in a schedule. Thus, for the case of real-time messages, the real-time MaxWeight algorithm becomes the maximal matching algorithm, giving priority to intraregion transmissions since this maximizes the number of links that can be simultaneously scheduled. Therefore, in order to maximize throughput, the real-time MaxWeight algorithm allocates service with no fairness considerations into account.

To explore the tradeoff between maximizing throughput and guaranteeing fairness, we measured the total network throughput for both algorithms, and the results are presented in Fig. 4. As it can be seen, for larger arrival rates the difference in both algorithms starts to increase since the real-time MaxWeight algorithm schedules more intraregion transmissions. Hence, in order to achieve proportional fairness we have to pay a price in terms of total network throughput.

Another way to explore the throughput-fairness tradeoff is by increasing the value of α. Note that $\alpha \to \infty$ corresponds to the limit case of max-min fairness, where the algorithm

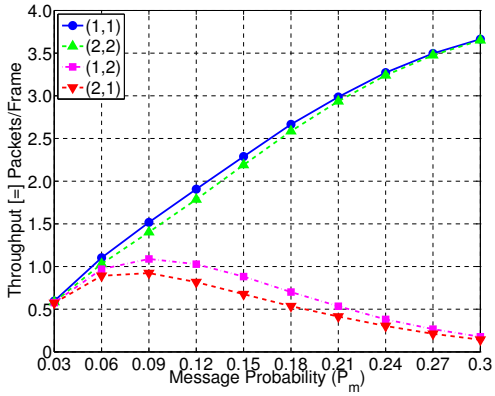

Fig. 3. Throughput for the Real-time MaxWeight Algorithm

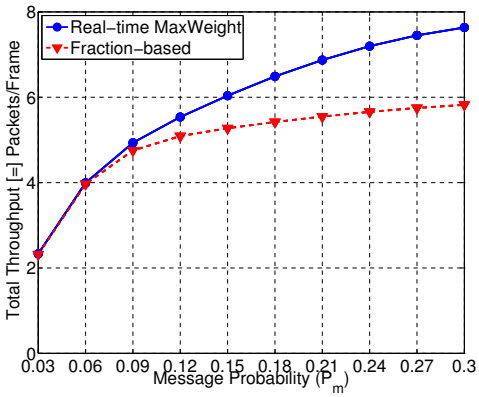

Fig. 4. Comparison of Total Network Throughput

tries to maximize the minimum fraction of service allocated to any given region pair. In Fig. 5 we plot the total network throughput when $P_m = 0.2$. As can be seen, the minimum throughput between different region pairs starts to increase with increasing α, but as can be observed in Fig. 6 we have to pay a price in terms of a small decrease in total network throughput, since we are sacrificing efficiency for fairness.

VII. Conclusions

In this paper we have studied the problem of service allocation and scheduling of real-time message requests under strict per-packet deadline constraints. We have presented an optimization framework that groups message requests by regions with similar interference and channel conditions, allowing us to design a solution that optimally allocates resources in the long term while meeting delay constraints. The solution allows for very general interference and arrival models. Using simulations we have showed the limitiations of previous approaches and why there is a need to develop a new solution to the problem we considered.

Appendix A
Proof of Lemma 1

We start proving Lemma 1 by presenting the following fact.

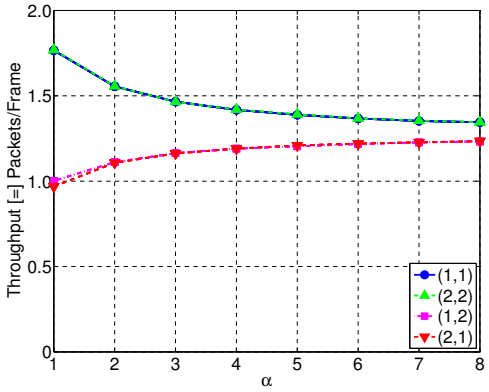

Fig. 5. Throughput for the Fraction-based Algorithm When α varies

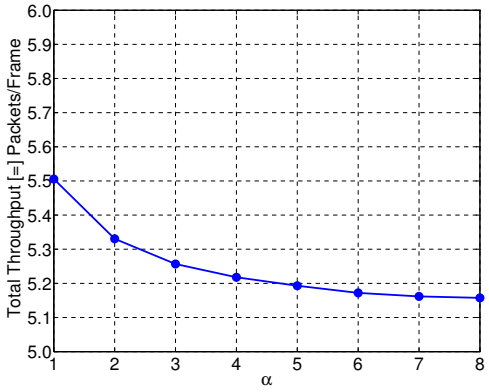

Fig. 6. Total Throughput for the Fraction-based Algorithm

Fact 1: The optimization in (5) can be performed over $\mathcal{S}(a(k),c(k))_{\mathcal{CH}}$, the convex hull of $\mathcal{S}(a(k),c(k))$; that is,

$$\max_{s\in\mathcal{S}(a(k),c(k))}\sum_{i,j\in\mathcal{M}}d_{ij}(k)\sum_{t\in\mathcal{T}}s_{ijt}=\max_{s\in\mathcal{S}(a(k),c(k))_{\mathcal{CH}}}\sum_{i,j\in\mathcal{M}}d_{ij}(k)\sum_{t\in\mathcal{T}}s_{ijt}.$$

The reason for this comes from the fact that the objective function is linear and therefore there must be an optimal point $\tilde{s}^*(a(k),c(k),d(k))\in\mathcal{S}(a(k),c(k))$. ◇

Proof of Lemma 1:

$$\begin{aligned}
&E\left[V(d(k+1))|d(k)=d\right]-V(d)\\
=&E\left[\frac{1}{2}\sum_{i,j\in\mathcal{M}}\{[d_{ij}+\tilde{a}_{ij}(k)-\tilde{I}^*_{ij}(a(k),c(k),d)]^+\}^2\right]\\
&-\frac{1}{2}\sum_{i,j\in\mathcal{M}}d_{ij}^2\\
\le&E\left[\frac{1}{2}\sum_{i,j\in\mathcal{M}}[d_{ij}+\tilde{a}_{ij}(k)-\tilde{I}^*_{ij}(a(k),c(k),d)]^2\right]\\
&-\frac{1}{2}\sum_{i,j\in\mathcal{M}}d_{ij}^2\\
=&E\left[\sum_{i,j\in\mathcal{M}}d_{ij}[\tilde{a}_{ij}(k)-\tilde{I}^*_{ij}(a(k),c(k),d)]\right.\\
&\left.+\frac{1}{2}\sum_{i,j\in\mathcal{M}}[\tilde{a}_{ij}(k)-\tilde{I}^*_{ij}(a(k),c(k),d)]^2\right]\\
\le&E\left[\sum_{i,j\in\mathcal{M}}d_{ij}\tilde{a}_{ij}(k)-d_{ij}\tilde{I}^*_{ij}(a(k),c(k),d)\right.\\
&\left.+\frac{1}{2}\sum_{i,j\in\mathcal{M}}\tilde{a}_{ij}^2(k)+a_{ij}^2(k)\right] \qquad (8)\\
\le&E\left[\sum_{i,j\in\mathcal{M}}d_{ij}\tilde{a}_{ij}(k)-d_{ij}\tilde{I}^*_{ij}(a(k),c(k),d)+a_{ij}^2(k)\right] \qquad (9)\\
=&B_1+E\left[\sum_{i,j\in\mathcal{M}}d_{ij}a_{ij}(k)\tilde{q}^*_{ij}(a(k),d)\right.\\
&\left.-d_{ij}\tilde{I}^*_{ij}(a(k),c(k),d)\right]\\
=&B_1-E\left[\sum_{i,j\in\mathcal{M}}\frac{1}{\epsilon}U_{ij}(\tilde{q}^*_{ij}(a(k),d))-d_{ij}a_{ij}(k)\tilde{q}^*_{ij}(a(k),d)\right.\\
&\left.+d_{ij}\tilde{I}^*_{ij}(a(k),c(k),d)-\frac{1}{\epsilon}U_{ij}(\tilde{q}^*_{ij}(a(k),d))\right]
\end{aligned}$$

where (8) and (9) follow from the definition of $\tilde{I}^*_{ij}(a(k),c(k),d)$ and $\tilde{a}_{ij}(k)$, respectively, and

$$B_1=\sum_{i,j\in\mathcal{M}}\lambda_{ij}^2+\sigma_{ij}^2.$$

From the definition of $\mathcal{C}$, $\tilde{\mu}\in\mathcal{C}$ implies that there exist $\tilde{\mu}(a,c)\in\mathcal{C}(a,c)$ for all a, c and $\tilde{\mu}_{ij}=E[\tilde{\mu}_{ij}(a,c)]$ for all $i,j\in\mathcal{M}$. For the rest of the proof we define $\tilde{\mu}_{ij}(a,c)$ to be such set of values associated to $\tilde{\mu}$. Thus:

$$\begin{aligned}
&E\left[V(d(k+1))|d(k)=d\right]-V(d)\\
\le&B_1-E\left[\sum_{i,j\in\mathcal{M}}\frac{1}{\epsilon}U_{ij}(\tilde{q}_{ij})-d_{ij}a_{ij}(k)\tilde{q}_{ij}\right. \qquad (10)\\
&\left.+d_{ij}\tilde{\mu}_{ij}(a(k),c(k))-\frac{1}{\epsilon}U_{ij}(\tilde{q}^*_{ij}(a(k),d))\right]\\
=&B_1-\sum_{i,j\in\mathcal{M}}d_{ij}(\tilde{\mu}_{ij}-\lambda_{ij}\tilde{q}_{ij})\\
&-\frac{1}{\epsilon}\sum_{i,j\in\mathcal{M}}\left\{U_{ij}(\tilde{q}_{ij})-E\left[U_{ij}(\tilde{q}^*_{ij}(a(k),d))\right]\right\}\\
\le&B_1-B_2\sum_{i,j\in\mathcal{M}}d_{ij}\\
&-\frac{1}{\epsilon}\sum_{i,j\in\mathcal{M}}\left\{U_{ij}(\tilde{q}_{ij})-E\left[U_{ij}(\tilde{q}^*_{ij}(a(k),d))\right]\right\}
\end{aligned}$$

where (10) follows from the fact that $\tilde{I}^*_{ij}(a(k), c(k), d)$ and $\tilde{q}^*_{ij}(a(k), d)$ are the solutions to (5) and (6), respectively, and Fact 1. Furthermore,

$$B_2 = \min_{i,j \in \mathcal{M}} \{\tilde{\mu}_{ij} - \lambda_{ij}\tilde{q}_{ij}\}.$$

■

APPENDIX B
PROOF OF THEOREM 1

From Lemma 2 we have

$$\begin{aligned}
&\frac{1}{\epsilon} \sum_{i,j \in \mathcal{M}} \left\{U_{ij}(q^*_{ij}) - E\left[U_{ij}(\tilde{q}^*_{ij}(a(k), d))\right]\right\} \\
\leq & B_1 - B_2 \sum_{i,j \in \mathcal{M}} d_{ij} - E\left[V(d(k+1))|d(k) = d\right] + V(d) \\
\leq & B_1 - E\left[V(d(k+1))|d(k) = d\right] + V(d).
\end{aligned}$$

The last inequality follows from the fact that $B_2 \sum_{i,j \in \mathcal{M}} d_{ij} \geq 0$. Taking expectations we obtain

$$\begin{aligned}
&\frac{1}{\epsilon} \sum_{i,j \in \mathcal{M}} \left\{U_{ij}(q^*_{ij}) - E\left[U_{ij}(\tilde{q}^*_{ij}(a(k), d(k)))\right]\right\} \\
\leq & B_1 - E\left[V(d(k+1))\right] + E\left[V(d(k))\right].
\end{aligned}$$

If we add the terms for $k = \{1, \ldots, K\}$ and divide by K we obtain

$$\begin{aligned}
&\frac{1}{\epsilon} \sum_{i,j \in \mathcal{M}} \left\{U_{ij}(q^*_{ij}) - E\left[\frac{1}{K}\sum_{k=1}^{K} U_{ij}(\tilde{q}^*_{ij}(a(k), d(k)))\right]\right\} \\
\leq & B_1 - \frac{E\left[V(d(K+1))\right]}{K} + \frac{E\left[V(d(1))\right]}{K} \\
\leq & B_1 + \frac{E\left[V(d(1))\right]}{K},
\end{aligned}$$

where the last inequality follows from the fact that the Lyapunov function is non-negative.

Using Jensen's inequality [10] we get the following

$$\begin{aligned}
&\frac{1}{\epsilon} \sum_{i,j \in \mathcal{M}} \left\{U_{ij}(q^*_{ij}) - U_{ij}\left(E\left[\frac{1}{K}\sum_{k=1}^{K} \tilde{q}^*_{ij}(a(k), d(k))\right]\right)\right\} \\
\leq & \frac{1}{\epsilon} \sum_{i,j \in \mathcal{M}} \left\{U_{ij}(q^*_{ij}) - E\left[\frac{1}{K}\sum_{k=1}^{K} U_{ij}(\tilde{q}^*_{ij}(a(k), d(k)))\right]\right\} \\
\leq & B_1 + \frac{E\left[V(d(1))\right]}{K}.
\end{aligned}$$

Taking the limit as $K \to \infty$, and assuming $E\left[V(d(1))\right] < \infty$, we get

$$\begin{aligned}
\limsup_{K \to \infty} \sum_{i,j \in \mathcal{M}} &\left\{U_{ij}(q^*_{ij})\right. \\
&\left. - U_{ij}\left(E\left[\frac{1}{K}\sum_{k=1}^{K} \tilde{q}^*_{ij}(a(k), d(k))\right]\right)\right\} \leq B\epsilon,
\end{aligned}$$

where $B = B_1$. ■

REFERENCES

[1] P. van de Ven, S. Borst, and S. Shneer, "Instability of maxweight scheduling algorithms," in *IEEE INFOCOM*, Rio de Janeiro, Brazil, Apr. 2009, pp. 1701 – 1709.

[2] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Autom. Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.

[3] ——, "Dynamic server allocation to parallel queues with randomly varying connectivity," *IEEE Trans. Inf. Theory*, vol. 39, pp. 466–478, Mar. 1993.

[4] B. Sadiq and G. de Veciana, "Throughput optimality of delay-driven maxweight scheduler for a wireless system with flow dynamics," in *Proc. Ann. Allerton Conf. Communication, Control and Computing*, Monticello, IL, USA, Oct. 2009.

[5] S. Liu, L. Ying, and R. Srikant, "Throughput-optimal opportunistic scheduling in the presence of flow-level dynamics," in *IEEE INFOCOM*, San Diego, CA, USA, Mar. 15–19, 2010.

[6] ——, "Scheduling in multichannel wireless networks with flow-level dynamics," in *ACM SIGMETRICS*, New York, NY, USA, Jun. 14–18, 2010, pp. 191–202.

[7] R. Laroia. (2010) Future of wireless? The proximate internet. [Online]. Available: http://www.cedt.iisc.ernet.in/people/kuri/Comsnets/Keynotes/Keynote-Rajiv-Laroia.pdf

[8] J. J. Jaramillo, R. Srikant, and L. Ying, "Scheduling for optimal rate allocation in ad hoc networks with heterogeneous delay constraints," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 5, pp. 979–987, May 2011.

[9] D. G. Luenberger, *Linear and Nonlinear Programming*, 2nd ed. Norwell, MA: Kluwer Academic Publishers, 2003.

[10] S. Boyd and L. Vandenberghe, *Convex Optimization*, 1st ed. New York, NY: Cambridge University Press, Mar. 2004.

[11] J. J. Jaramillo and R. Srikant, "Optimal scheduling for fair resource allocation in ad hoc networks with elastic and inelastic traffic," in *IEEE INFOCOM*, San Diego, CA, USA, Mar. 15–19, 2010.

[12] M. J. Neely, E. Modiano, and C.-P. Li, "Fairness and optimal stochastic control for heterogeneous networks," in *IEEE INFOCOM*, vol. 3, Miami, FL, USA, Mar. 13–17, 2005, pp. 1723–1734.

[13] A. Stolyar, "Maximizing queueing network utility subject to stability: Greedy primal-dual algorithm," *Queueing Systems*, vol. 50, no. 4, pp. 401–457, Aug. 2005.

[14] A. Eryilmaz and R. Srikant, "Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control," in *IEEE INFOCOM*, vol. 3, Miami, FL, USA, Mar. 13–17, 2005, pp. 1794–1803.