Opportunistic Scheduling for Wireless Systems with Multiple Interfaces and Multiple Constraints

Sunil Suresh Kulkarni
sunilkul@ecn.purdue.edu

Catherine Rosenberg
cath@ecn.purdue.edu

School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN-47907-2035, USA

ABSTRACT

We consider a scheduling problem for packet based systems with time-varying channel conditions. Designing scheduling mechanisms that take advantage of time-varying channel conditions, which are different for different users, is necessary to improve system performance; however this has to be done in a way that provides some level of fairness among the users. Such scheduling mechanisms are termed as opportunistic. We generalize the opportunistic scheduling mechanisms in the literature on two fronts. First, we formulate and solve an opportunistic scheduling problem with multiple general long term QoS constraints and a general system objective function. We call the solution of this opportunistic scheduling problem as a *generalized water filling* solution. Then, we generalize this problem to include multiple interface systems in which several users can be served simultaneously. Apart from the long term QoS constraints specified by each user, multiple interface systems are constrained with other physical limitations imposed by the system structure. Our main contribution is to show that the structure of the optimal opportunistic scheduling policy is carried over to the problem with general constraints and multiple interfaces. We also study the stability of the multiple interface systems and propose a throughput optimal scheduling rule for such systems.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: Modeling Techniques; C.2.1 [**Computer Communication Networks**]: Network Architecture and Design—*Wireless Communication*; G.3 [**Probability and Statistics**]: Stochastic Processes

General Terms

QoS Provisioning, Performance evaluation and modeling, Optimization, Theory

Keywords

Opportunistic scheduling, QoS scheduling, Multiple Interface Systems

1. INTRODUCTION

Wireless channels, in contrast to their wireline counterparts have time-varying location-dependent characteristics and different wireless users experience different channel conditions at a given time. These channels are affected by user shadowing and path losses due to changing environments and due to user mobility. In CDMA based systems, wireless channels are affected by the co-channel interference due to other users. Also in the case of satellite systems, the channel conditions vary due to the weather conditions and satellite movements. In prior works [10, 4], it has been argued that the variations in the channel conditions should be exploited to increase the system throughput. The basic idea behind exploiting the channel variations is to schedule a user having the best channel condition at a given time. Such scheduling mechanisms are called *Opportunistic Scheduling Mechanisms.* If the service requirements of all the users are flexible, such opportunistic scheduling methods can result in reduced interference, higher spectrum utilization and increased system throughput. CDMA-HDR (IS-856) is an example of a high data rate system that can take advantage of time-varying channel conditions through the use of an opportunistic scheduling mechanism.

The problem of packet scheduling for systems with time-varying channel conditions can be illustrated as follows. (see Figure 1). Consider a base station with fixed transmission power and a packet based downlink scheduling mechanism. The wireless channel for each user differs depending on the location, the surrounding environment, and mobility. Assume that each user reports its downlink channel condition to the base station in a periodic fashion. Thus the base station knows exactly the channel condition and hence the data rate it can offer to each user on the downlink channel if only that user is served at a given time. After finishing a packet transmission, the base station must choose the next (single) user to whom it will send the next packet. It will use the transmission rate as determined by that user's channel condition. Such systems are called *single interface systems.*

There are systems of interest which cannot be modeled in the above framework. Such systems can be described as

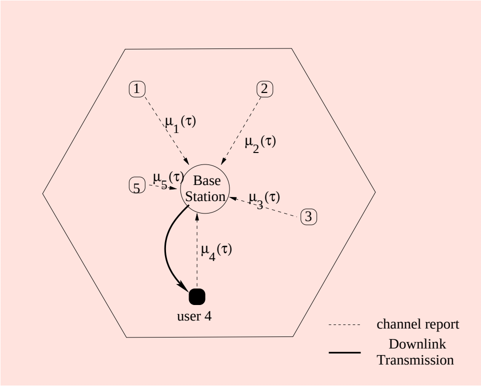

Figure 1: Opportunistic Scheduling: A typical wireless base station downlink

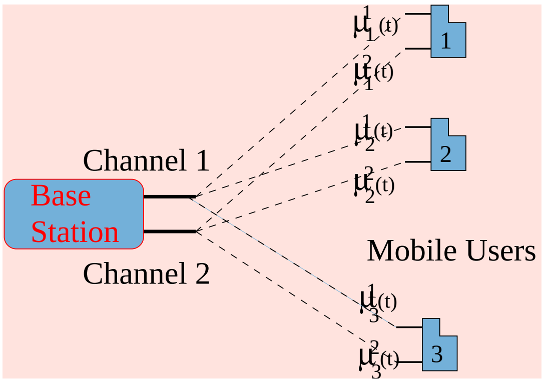

Figure 2: Multiple Interface Opportunistic Scheduling

multiple interface systems where the user channel conditions are time-varying over all the interfaces and the base station has the freedom to serve several users on several interfaces simultaneously. For example, in multi-carrier CDMA systems every user has a time-varying channel on each carrier. In multiple-antenna systems, multiple users can be served on different antennas. In OFDM systems, fading and shadowing are different for each carrier. In multiple beam satellite systems, a satellite can transmit to different ground terminals over different channels. Depending on the specifics of the system, the number of interfaces can vary from two to more than thousand.

To describe the scheduling problem in such systems, again consider a packet based downlink scheduling (see Figure 2). All the transmissions on different interfaces start and end at the same time. Each user has different channel conditions over different interfaces. Assume that each user reports its channel condition on every interface to a single base station. The base station must schedule several users for transmission over multiple interfaces in a way that satisfies the physical constraints. The physical constraints can differ from system to system taking into account the different technologies used at the base station, but these physical constraints must be satisfied at all times. We do not assume that the different interfaces use the same technology, but we do assume that the base station has complete knowledge of the current channel state for each user on each interface. This excludes the case in which each interface is managed by a different base station. It is not necessary that all users have all the interfaces. If a particular user does not support a particular kind of interface, the data transmission rate for that user over that kind of interface is always assumed to be zero. Thus, without loss of generality we assume that all users support all interfaces.

This paper is organized in the following way. In Section 2, we discuss previous relevant work in this area. In Section 3, we formulate our multiple constraint opportunistic scheduling problem over a single interface and provide a generalized water-filling solution for this problem. We also discuss relevant system considerations in the same section. Section 4 formulates the opportunistic scheduling problem over multiple interfaces and provides a similar water-filling solution for it. For opportunistic scheduling over multiple interfaces Section 5 provides a throughput optimal scheduling rule that guarantees stable user queues. We conclude in Section 6.

2. PREVIOUS WORK

The problem of exploiting the channel state variations to increase the system throughput has been in focus in recent years. The typical representative term used to describe the performance gain due to these channel state variations across different users over time, is *multi-user diversity*. Multi-user diversity has a similar effect to that of a space or time diversity, enabling some improvement in the system performance. This multi-user diversity has been described from an information theoretic viewpoint in [19] and the references contained therein. Specifically in that paper the authors describe a water-filling mechanism, which is asymptotically optimal in maximizing the system throughput. Water-filling power control has also been proposed for the CDMA forward power control problem (see [7] for more details). The water-filling method schedules the user with the highest data rate at a given time. A bias (different for different users) is added to the time-varying data rate for each user to equalize the differences between the service offerings due to such greedy selection. In [4] the authors present a water-filling scheduling scheme for the Qualcomm/HDR system which is also proportionally fair. In [20] the authors suggest to increase or even introduce random variations in the channel conditions using multiple dumb antennas so that the multi-user diversity can be fully exploited. They analyze such a system and give the asymptotic bounds for the total channel capacity. The main drawback of these approaches is that individual QoS requirements cannot be taken into account in designing a scheduling policy.

In [14, 16] the authors extend wireline scheduling policies to wireless networks and present wireless fair scheduling policies which give short term and long term fairness bounds. While this approach provides fairness guarantees, it assumes that the channel quality is either good or bad. In [12] the authors present scheduler called WCFQ (Wireless Credit-based Fair Queuing) to provide (only) temporal fairness among the users with statistical fairness bounds. Their approach is based on CBFQ (Credit Based Fair Queuing), a scheduler for wired systems. WCFQ trades off the fairness and throughput to exploit the channel time variations by mapping channel condition into to cost function. In [17] the authors present and study a practical scheme to exploit channel time variations in 802.11 based ad-hoc networks. While maintaining same level of fairness, they provide a modification in 802.11 protocol to obtain a better throughput than that of plain 802.11 protocol.

In [1] the authors discuss the scheduling problem with fixed deadlines and prove that the greedy algorithm performs no worse than half of offline optimal schedule in terms revenue. This approach considers the problem on a packet by packet basis and hence we feel that this approach is inappropriate for the real wireless scheduling systems.

In [2, 18, 6] the authors study throughput optimal scheduling rules. If there exists any scheduling rule which can guarantee stable queues then a throughput optimal scheduling rule also guarantees stable queues. Thus, throughput optimal scheduling rules have the maximum system stability region in the space defined by the average incoming rates. Throughput optimal scheduling rules in time-invariant case are studied in [3]. In [15] the authors study a specific case of multi-beam satellite systems, suggesting a throughput optimal scheduling rule and they extend these results for wireless networks. In [11] and [10] the authors introduce three different QoS constraints, which they call Resource Sharing Fairness Constraint, Performance Based Fairness Constraint and Minimum Performance Fairness Constraint. They provide optimal opportunistic scheduling mechanism for these special constraints. In [18] the authors present a technique to extend the throughput optimal scheduling rules to guarantee the minimum data rate constraint. Most of these results except [13] consider only the single interface case. Specifically, it is inherently assumed that only one user will be scheduled at a given time. In [13] the authors consider a throughput maximization problem with deterministic and probabilistic GPS-like fairness constraint for multiple interface systems. But their approach is limited by specific objective and constraints. Also, most of the works concentrate on a single form of fairness constraint.

Our work which considers multiple constraints and multiple interfaces unifies many of the results and puts all of them into a single framework. Thus the main contributions of this paper are as follows.

- For general multiple long term QoS constraints, we show that the optimal scheduling policy for a single interface system has a *argmax* solution. We call this solution structure a generalized water-filling solution.
- We show that the scheduling policy for multiple interface systems with general long term QoS constraints also has the similar generalized water-filling structure.
- For multiple interface systems, we prove that a Modified Largest Weighted Work First (M-LWWF) scheduling policy is throughput optimal (under a Markovian state model).

3. OPPORTUNISTIC SCHEDULING OVER A SINGLE INTERFACE

To formulate the scheduling problem for a time-varying channel over a single interface with multiple QoS constraints, we make the following simplifying assumptions. We assume that the system operates on a timeslot by timeslot basis. The width of each timeslot is fixed and the channel conditions do not vary during a timeslot. In the real systems the physical frame transmission size can be selected from a finite discrete set. Hence depending on the current data rate, the transmission of a physical frame can take more than one timeslot to transmit a physical frame. However we assume that the physical frame transmission size can be varied according to the transmission rate so that transmissions begin and end exactly at timeslot boundaries. This is also assumed in [11, 10, 18, 2, 15, 19, 20]. We also assume that all the transmissions are successful. In real systems, the users send the necessary channel state information to the base station in a periodic fashion. Hence we assume that the base station knows the exact channel state and the exact data rate with which it can transmit to each user. We assume that at most one user can be served in a timeslot. We also assume that all the users are greedy and each user always has data to receive on the downlink.

3.1 Constraints and Objectives

The QoS requirements for different users can be different and each user can potentially specify its own requirements. In general, these requirements can be grouped into short term and long term constraints. In this paper we only consider the long term requirements because long term requirements are of more importance than the short term requirements for elastic applications. (For more details of opportunistic scheduling policies with short term fairness constraints please see [8].) The widely referred long term QoS constraints are of two different types.

- Processor sharing constraint: User i specifies a weight ϕ_i and expects to get at least a ϕ_i fraction of the server time. Thus this fairness constraint can be viewed as a fairness criterion similar to the generalized processor sharing. For a stable system, obviously we need $\sum_i \phi_i \leq 1$.
- Data Rate constraint: A user asks for a minimum data rate guarantee to sustain its applications. This is a more appealing QoS criterion from the user standpoint, as most of the non-elastic applications need some minimum data rate. However a user might request a high data rate but may experience very poor channel conditions. Hence, it is not easy to specify the feasibility of the system under general minimum performance constraints because we do not assume any knowledge of the channel conditions.

We have mentioned only few possible fairness or QoS requirements. But potentially there can be many more and combinations of those constraints can also be specified. For example, a wireless node may specify two constraints, one for maximum power consumption and other for minimum data rate requirement. The main constituent of the energy consumption is the energy consumed in radio electronics. To satisfy the maximum power consumption constraint, an upper limit on the processor time share can be calculated.

Thus rather than having a lower bound on the processor share (as in GPS), there could potentially be an upper bound on the processor share to reduce the power consumption. By putting an upper limit on the processor time share a mobile device can guarantee maximum power drain, which transforms into a minimum on time for this mobile device without replenishing its energy sources. Thus the two constraints in this scenario become maximum processor share constraint and minimum data rate constraint.

3.2 Notation

We start by introducing the notation and then state the general constraints associated with all the users. We use similar terminology and the notation as in reference [10].

- $\mathcal{N}$: This denotes the set of users, usually indexed by i. The users will be indexed from 1 to N.
- $\mu_i(t)$: This denotes the data rate for user i in timeslot t. Thus $\vec{\mu}(t) = [\mu_1(t), \cdots, \mu_N(t)]$ denotes the vector of the data rates for all users at time t.
- $\mathcal{K}_i$: This denotes the set of constraints for user i. The constraints are usually indexed by j.
- f_i, g_i^j: f_i denotes the utility function and g_i^j denotes the j^{th} constraint function associated with user i.
- $Q(\vec{\mu}(t))$: This denotes a policy to select a user to serve in timeslot t, given $\vec{\mu}(t)$.

We denote the indicator function by the letter I, thus $I_{Q(\mu(\vec{t}))=i}$ is 1 if in timeslot t user i is selected for service, otherwise zero. We assume that each user is only interested in long term QoS constraints. Then for user i, the j^{th} long term QoS requirement constraint is stated as follows. (We omit the timeslot variable t by some abuse of notation.)

$$\forall i \in \mathcal{N}, j \in \mathcal{K}_i \;\; E\{g_i^j(\mu_i) I_{Q(\vec{\mu})=i}\} \geq G_i^j$$

For the processor sharing constraint, the functions g_i^j can be taken as unit functions, $g_i^j(\mu_i(t)) = 1$. Thus the processor sharing constraint can be written as

$$\forall i, \;\; E\{I_{Q(\vec{\mu})=i}\} \geq G_i \equiv \phi_i$$

In words this constraint can be phrased as, on the long term a user i gets a base station time share greater than or equal to ϕ_i. Similarly for the data rate constraint g_i^j can be taken as $g_i^j(\mu_i(t)) = \mu_i(t)$ and hence the data rate constraint can be written as

$$\forall i, \;\; E\{\mu_i I_{Q(\vec{\mu})=i}\} \geq G_i \equiv R_i$$

We define a general system objective as follows.

$$max_Q \sum_{i \in \mathcal{N}} E\{f_i(\mu_i(t)) I_{Q(\vec{\mu}(t))=i}\}$$

Usually the objective of the system is to maximize the total system throughput. In that case the utility functions f_i can be taken as $f_i(\mu_i(t)) = \mu_i(t)$. Intuitively this would force the optimal policy to choose a user having a better channel (higher data rate) to maximize the system throughput.

3.3 Problem Formulation and the Solution

Thus the single interface scheduling problem can be formulated as an optimization problem as follows.

$$\begin{aligned} max_Q \qquad & \sum_{i \in \mathcal{N}} E\{f_i(\mu_i) I_{Q(\vec{\mu})=i}\} \quad (1) \\ such\ that\ \forall i \in \mathcal{N},\ \forall j \in \mathcal{K}_i \qquad & E\{g_i^j(\mu_i) I_{Q(\vec{\mu})=i}\} \geq G_i^j \end{aligned}$$

Let us denote an optimal policy by Q^*. Then we can show the following,

Theorem 1. *The solution of the above single interface constrained opportunistic scheduling problem, if one exists, is of the following form.*

$$Q^* \;=\; argmax_i\{f_i(\mu_i) + \sum_{j=1}^{\mathcal{K}_i} \lambda_i^j g_i^j(\mu_i)\} \qquad (2)$$

$$\exists \lambda_i^j \geq 0 \quad s.t. \quad E\{g_i^j(\mu_i) I_{Q(\vec{\mu})=i}\} > G_i^j \Rightarrow \lambda_i^j = 0$$

Proof. The proof follows directly by formulating the Lagrangian as done in [10]. Consider any feasible policy Q. Then there exist nonnegative constants λ_i^j such that the following holds. (We have removed the argument of the policy Q from the notation for simplicity. We denote the constraint set for user i by K_i.)

$$\begin{aligned} & \sum_{i \in \mathcal{N}} E\{f_i(\mu_i) I_{Q=i}\} \\ \leq & \sum_{i \in \mathcal{N}} E\{f_i(\mu_i) I_{Q=i}\} + \sum_{\substack{j \in \mathcal{K}_i \\ i \in \mathcal{N}}} \lambda_i^j \Big[E\{g_i^j(\mu_i) I_{Q=i}\} - G_i^j\Big] \\ = & \sum_{i \in \mathcal{N}} E\Big[\big(f_i(\mu_i) + \sum_{j \in \mathcal{K}_i} \lambda_i^j g_i^j(\mu_i)\big) I_{Q=i}\Big] - \sum_{\substack{j \in \mathcal{K}_i \\ i \in \mathcal{N}}} \lambda_i^j G_i^j \\ \leq & \sum_{i \in \mathcal{N}} E\Big[\big(f_i(\mu_i) + \sum_{j \in \mathcal{K}_i} \lambda_i^j g_i^j(\mu_i)\big) I_{Q^*=i}\Big] - \sum_{\substack{j \in \mathcal{K}_i \\ i \in \mathcal{N}}} \lambda_i^j G_i^j \\ = & \sum_{i \in \mathcal{N}} E\{f_i(\mu_i) I_{Q^*=i}\} + \sum_{\substack{j \in \mathcal{K}_i \\ i \in \mathcal{N}}} \lambda_i^j \Big[E\{g_i^j(\mu_i) I_{Q^*=i} - G_i^j\Big] \\ = & \sum_{i \in \mathcal{N}} E\{f_i(\mu_i) I_{Q^*=i}\} \end{aligned}$$

Thus we have proved that throughput of the policy defined in equation 2 is greater than or equal to that of any other policy. □

The constants λ's are the Karush-Kuhn-Tucker (KKT) multipliers and depend on the multidimensional distribution of $\mu_i(t)$'s. The optimal policy is always of an argmax type for the type of constraints and objectives we have defined. The arguments for the argmax are the weighted and shifted sums of functions of the current data rate for each user. If all the users specify only the processor sharing constraint, the optimal policy becomes $argmax_i\{\mu_i + \lambda_i\}$. Thus the optimal policy (with processor sharing constraints) adds a bias equal to the KKT multipliers to the data rate values resulting in the optimal *water-filling* solution. If all the users specify only a minimum data rate constraint, the optimal policy becomes $argmax_i\{\mu_i(1 + \lambda_i)\}$. Thus the optimal policy (with a data rate constraint) multiplies the data rate values with the KKT multipliers. These two special cases have been studied in [11]. In general, in the solution to the generalized opportunistic scheduling problem, we affine translate the functions of the data rates and then use an argmax rule to obtain the optimal policy. We call the general optimal solution a *generalized water-filling* solution.

An example of a single interface opportunistic scheduling problem with multiple constraints is as follows. Suppose that the objective of the opportunistic scheduling problem is to maximize the *goodput*. By goodput we mean that we consider the transmission of only user data and not any headers (overhead associated with the transmission). Assume that the header size is of H bytes per transmission. Also all the users specify two constraints. The first constraint is the usual processor sharing constraint with weight of ϕ_i for user i, and second is a minimum goodput rate constraint (*i.e.*, data rate without headers) of R_i. Thus the problem can be formulated as follows.

$$\begin{array}{ll} max_Q & \sum_{i\in\mathcal{N}} E\{(\mu_i - H)I_{Q(\vec{\mu})=i}\} \\ s.t.\ \forall i, & E\{I_{Q(\vec{\mu})=i}\} \geq \phi_i;\ \ E\{(\mu_i - H)I_{Q(\vec{\mu})=i}\} \geq R_i \end{array}$$

The solution of this problem is,

$$\begin{array}{rcl} Q^* & = & argmax_i\{\mu_i + \lambda_i^1 + \lambda_i^2(\mu_i - H)\} \\ i.e.,\ \ Q^* & = & argmax_i\{\mu_i(1+\lambda_i^2) + \lambda_i^1 - H\lambda_i^2\} \end{array}$$

The λ's are the KKT multipliers as defined in Theorem 1. Thus the optimal scheduling policy is again an argmax policy of the affine translated data rate values.

We now discuss various issues related to the optimal solution of the opportunistic scheduling problem.

3.4 Discussion

The first issue is how to calculate the KKT multipliers in a real system so that we obtain the desired optimal scheduling policy. In [11, 10] the authors present a stochastic approximation algorithm to calculate such constants. The basic idea behind these algorithms is to start with all zero KKT constants. Then in each timeslot, along a sample path, a correction term proportional to the error term (the difference between the actual constraint bound G_i^j and the so far achieved average constraint value) is added. A similar approach can be taken to calculate the $\lambda's$ in the equation 2. We plan to use similar stochastic approximation algorithm to calculate these constant in our future work. Such algorithms are not computationally complex and hence can be used in each timeslot. The results of [11, 10] show that these algorithms converge quite fast.

Note that if the system is unfeasible then the stochastic approximation method will not converge. Thus we must guess the feasibility heuristically if the stochastic approximation algorithm seems to diverge. In fact, in this sense the water-filling solution is limited because it does not give any information about the system feasibility. Also after an arrival of a new user or after a departure of a user all the KKT multipliers change and must be recalculated.

The optimal solution can generate a tie between two (or more) users. If the probability distributions of the data rate values are continuous (assuming the functions f_i, g_i are one to one), ties can be broken with any randomization rule. But if the data rate values have a discrete distributions then we must specify a randomization rule to break the ties so as to satisfy the constraints. A simple approach of stochastic approximation will lead to oscillations around the constants in the optimal solution. In such tie cases, a heuristic technique for obtaining the randomization probabilities can be developed. We note that in practical systems the data rate values are discrete and hence there is a need for the randomization probabilities.

The solution for the opportunistic scheduling assumes that all the users are greedy and hence the solution methodology is good for the heavy traffic scenario. Also note that the optimal solution with the processor sharing constraints does not guarantee the *exact* generalized processor sharing (which guarantees distribution of excess server capacity among the active users in proportion to their weights) if some users are inactive. Also other types of guarantees, *e.g.,* short term fairness constraints and buffer overflow constraints cannot be provided with this water-filling approach.

Finally note that we have assumed that each transmission takes only one timeslot but in actual systems the physical frame transmission can take multiple timeslots. Thus, the practical problem is more complex.

4. OPPORTUNISTIC SCHEDULING OVER MULTIPLE INTERFACES

The next generation of wireless communication devices will be equipped with more than one interface. Each interface provides different transmission characteristics. Such devices have been already discussed in the literature but within a context of connectivity and mobility. A device may have more than one interface because a single technology might not be sufficient in terms of coverage or because in different environments different technologies might be more suitable. We consider a general opportunistic scheduling problem over multiple interfaces with long term user QoS constraints (similar to the QoS constraints described in previous section) and physical constraints. The physical constraints are imposed by the system structure and must be satisfied in each timeslot. For example, consider the different interfaces as different physical antennas. There are K physical interfaces (antennas) for each user. The communication bandwidth is divided in K bands and each antenna can be tuned to any band. Thus each antenna can be used for any interface, but an antenna can be used for only one interface at a given time. Thus in the above case, a physical constraint can be specified as follows. If an interface k is assigned to user i in a given timeslot, then another interface l cannot be assigned to user i and another user j cannot be assigned to the interface k in the same timeslot. This is precisely the scenario in the multi-beam satellite systems. In such systems, there are K beams and N earth based stations. A beam can be used to serve any one of the stations in a given timeslot. At most one beam could be used to serve a station. A more complex example of physical constraint can be a modified version of the above physical constraint as follows. Suppose that the base station is limited by the maximum total power it can use for the transmission in any given timeslot. In this example, the base station has to choose a power distribution among the different interfaces such that the total transmitted power must be less than the maximum allowed power consumption. (For example, power assignment problem for up-link scheduling in CDMA systems is considered in [9] and power assignment problem in multi-beam satellite networks is considered in [15].)

We denote by $\mathcal{K}$ the set of interfaces and index the interfaces by the letter k. Thus $\vec{\mu_i} = (\mu_i^1, \cdots, \mu_i^K)$ and $[\mu]$ is the $N \times K$ matrix where μ_i^k denotes the current data rate for user $i \in \mathcal{N}$ over interface $k \in \mathcal{K}$. For notational convenience only, we assume that each user has only one QoS constraint (referenced by g_i). A stationary policy $\vec{Q} = (Q^1, \cdots, Q^K)$ denotes a vector function on $[\mu]$, which assigns each interface to a particular user in each timeslot according to the specified rule for Q. Though other physical constraints can also be taken into account in a similar fashion, in the following discussion we assume the following physical constraint: any feasible policy cannot assign two users to the same interface or two interfaces to the same user in any timeslot, *i.e.,* $k \neq \hat{k} \Leftrightarrow Q^k([\mu]) \neq Q^{\hat{k}}([\mu])$.

The multiple interface opportunistic scheduling problem can be defined as follows.

$$\begin{array}{rl} max_Q & \displaystyle\sum_{i\in\mathcal{N}}\sum_{k\in\mathcal{K}} E\{f_i(\mu_i^k) I_{Q^k([\mu])=i}\} \qquad (3) \\ such\ that\ \forall i & \displaystyle\sum_{k\in\mathcal{K}} E\{g_i(\mu_i^k) I_{Q^k([\mu])=i}\} \geq G_i \\ if\ \ k \neq \hat{k} \ \ \Leftrightarrow & Q^k([\mu]) \neq Q^{\hat{k}}([\mu]) \end{array}$$

Now define a function $Kargmax_{i,k}(f_{i,k})$ which is a solution of the following optimization problem.

$$\begin{array}{rl} max_{a_{i,k}} & \displaystyle\sum_{i\in\mathcal{N}}\sum_{k\in\mathcal{K}} E\{a_{i,k} f_{i,k}\} \qquad (4) \\ s.t.\ a_{i,k} \in \{0,1\}, & k \neq \hat{k} \Leftrightarrow a_{i,k} + a_{i,\hat{k}} \leq 1 \\ & i \neq \hat{i} \Leftrightarrow a_{i,k} + a_{\hat{i},k} \leq 1 \end{array}$$

In words, $Kargmax_{i,k}(f_{ik})$ is a function defined as follows: Choose at the most K entries from the matrix $[f_{ik}]$ of $N \times K$ entries, satisfying the physical constraints, such that the sum of the selected terms is maximum. For the constraints we have considered, we choose a maximum of one entry per column per row. Hence, this special case of $Kargmax_{i,k}$ is a weighted bipartite graph matching problem (or a maximal weight matching problem). The nodes from one part of the graph represent users and the nodes from another part represent the interfaces. The weights of each link is specified by the the term $f_{i,k}$.

Let us denote an optimal policy by $\vec{Q^*}$. Then we claim that,

Theorem 2. *The solution of the multiple interface constrained opportunistic scheduling problem defined in equation 3, if one exists, is of the following form.*

$$\begin{array}{rcl} \vec{Q^*} & = & Kargmax_{i,k}\{f_i(\mu_i^k) + \lambda_i g_i(\mu_i^k)\} \qquad (5) \\ \exists\lambda_i \geq 0 & s.t. & \displaystyle\sum_{k\in\mathcal{K}} E\{g_i(\mu_i^k) I_{Q^k([\mu])=i}\} > G_i \Rightarrow \lambda_i = 0 \end{array}$$

Where, the $Kargmax_{i,k}(f_{ik})$ is a function defined as above.

Proof. Consider any feasible policy Q. Then there exist nonnegative constants λ_i such that the following holds.

$$\begin{array}{l} \displaystyle\sum_{\substack{i\in\mathcal{N}\\k\in\mathcal{K}}} E\{f_i(\mu_i^k) I_{Q^k=i}\} \\ \quad \begin{array}{cl} \leq & \displaystyle\sum_{\substack{i\in\mathcal{N}\\k\in\mathcal{K}}} E\{f_i(\mu_i^k) I_{Q^k=i}\} + \sum_{\substack{i\in\mathcal{N}\\k\in\mathcal{K}}} \lambda_i \Big[E\{g_i(\mu_i^k) I_{Q^k=i}\} - G_i\Big] \\ = & \displaystyle\sum_{\substack{i\in\mathcal{N}\\k\in\mathcal{K}}} E\Big[\big(f_i(\mu_i^k) + \lambda_i g_i(\mu_i^k)\big) I_{Q^k=i}\Big] - \sum_{\substack{i\in\mathcal{N}\\k\in\mathcal{K}}} \lambda_i G_i \\ \leq & \displaystyle\sum_{\substack{i\in\mathcal{N}\\k\in\mathcal{K}}} E\Big[\big(f_i(\mu_i^k) + \lambda_i g_i(\mu_i^k)\big) I_{Q^{*k}=i}\Big] - \sum_{\substack{i\in\mathcal{N}\\k\in\mathcal{K}}} \lambda_i G_i \qquad (6) \\ = & \displaystyle\sum_{\substack{i\in\mathcal{N}\\k\in\mathcal{K}}} E\{f_i(\mu_i^k) I_{Q^{*k}=i}\} + \sum_{\substack{i\in\mathcal{N}\\k\in\mathcal{K}}} \lambda_i \Big[E\{g_i(\mu_i^k) I_{Q_{*k}=i}\} - G_i\Big] \\ = & \displaystyle\sum_{\substack{i\in\mathcal{N}\\k\in\mathcal{K}}} E\{f_i(\mu_i^k) I_{Q^{*k}=i}\} \end{array} \end{array}$$

Equation 6 follows from the the definition of $\vec{Q^*}$ in equation 5 and by the property of $Kargmax$ function defined in equation 4. □

Thus the same generalized water-filling structure is carried over to the optimal policy for multiple interface opportunistic scheduling problem. The solution can be considered as a multidimensional water-filling solution. We note that even when the constants in the optimal solution are given, $Kargmax$ solves a weighted bipartite graph matching problem in every timeslot. Even for simple constraints the optimal policy for the multiple interface opportunistic scheduling problem needs complex computations in each timeslot. As opposed to this, the optimal policy for the single interface opportunistic scheduling problem simply selects the largest entry. In a special case where each user has the same data rate over all the interfaces in every timeslot (different users still have different data rates) the optimal policy is of the form $Kargmax_i(f_i(\mu_i) + \lambda_i g_i(\mu_i))$, where $Kargmax$ becomes a function which simply selects K highest entries. We note that for different physical constraints, $Kargmax$ is a different function and an optimization problem in itself. For example, for the physical constraints such as power allocation constraints, the $Kargmax$ is a function which chooses a power allocation strategy in each timeslot such that the sum of the selected $f_i(\mu_i^k) + \lambda_i g_i(\mu_i^k)$ terms is maximum. In such cases, we propose to use two approaches to solve the $Kargmax$ optimization problem and compare the results in our future work. The first approach is the optimal, computationally complex solution of $Kargmax$ optimization problem and the second is a suboptimal, computationally simple solution of actual $Kargmax$ optimization problem. The constants $\lambda's$ will be calculated with stochastic approximation algorithm as in previous single interface problem.

5. THROUGHPUT OPTIMAL SCHEDULING RULE FOR MULTIPLE INTERFACE

Stability of the system is the most important issue for any system design. By stability we mean that the queue length should remain finite. Intuitively this can be ensured if the service rate is greater than the incoming traffic rate for each queue. In simple static queuing systems this translates into ensuring that each user gets a enough time share of the server yielding a service rate greater than the incoming user traffic rate. But in time-varying dynamic systems, ensuring only this is not enough as there is no simple linear relation between the amount of the processor share and the corresponding service rate a user gets. Hence, it is also important to *optimally time* the service offering when the channel conditions are better for a user. Such stable policies might depend on the average incoming traffic rates and any method which requires the exact incoming rates to be known to obtain a stable scheduling policy critically depends on estimating the incoming rates. Throughput optimal scheduling polices on the other hand, are the policies which guarantee stability under any incoming traffic rates, if at all there is a stable scheduling policy.

For the single interface case a throughput optimal scheduling rules has been given in [2]. In this section we generalize the throughput optimal scheduling rules termed as M-LWWF, *i.e.,* Modified Largest Weighted Work First. This rule for the single interface case is stated as follows. In any given timeslot, serve a user having maximum $a_i q_i(t)\mu_i(t)$. Here, a_i's are any positive (nonzero) constants, $q_i(t)$ is the queue size for a user i at time t. Thus $Q^* = argmax_i\{a_i q_i \mu_i\}$ is a throughput optimal scheduling rule for the single interface case.

For the multiple interface case we generalize this theorem as follows.

Theorem 3.

$$\vec{Q^*}(t) = Kargmax_{i,k}\{a_i q_i(t)\mu_i^k(t)\} \tag{7}$$

is a throughput optimal scheduling rule, i.e., *if there exist any scheduling rule that makes all user queues stable then the above rule also makes all user queues stable. (Kargmax is a function which chooses at most K terms satisfying the physical constraints, such that the sum of all of those terms is maximum.)*

Proof. Refer to the Appendix. □

Note that we do not assume that the users are greedy as we have assumed in Theorems 1 and 2. For more details on throughput optimal rules refer to [3, 18, 15]. This rule can be used to guarantee some simple user QoS constraints such as data rate constraints. A virtual token queue with a token generation rate of R_i can be implemented. Then the throughput optimal scheduling policy applied to this virtual token queue process guarantees a data rate R_i to a user i. For a tie situations, we do not need to specify the randomization probabilities as any randomization method will be throughput optimal.

6. CONCLUSIONS

In this paper, we studied the scheduling problem with different time-varying channel conditions for each user. We generalized the opportunistic scheduling problem to include multiple general user QoS requirements and showed that the *argmax* policy with a water-filling method provides an optimal solution. We then studied the opportunistic scheduling problem for multiple interface systems with user QoS constraints and the physical constraints imposed by the system structure. We showed that the optimal policy for this problem is also of the *argmax* and water-filling nature. In our future work, we plan to simulate and study above algorithms for the realistic wireless channel conditions. For multiple interface systems we plan to develop efficient algorithms to estimate the optimal policy-constants using multidimensional stochastic approximation algorithms. We also generalized the throughput optimal scheduling rule for the multiple interface systems with general physical constraints via fluid limit techniques.

7. ACKNOWLEDGMENTS

This work was supported in part by the Indiana Twenty-First Century Fund through the Indiana Center for Wireless Communication and Networking.

8. REFERENCES

[1] M. Agarwal and A. Puri. Base station scheduling of requests with fixed deadlines. In *Proceedings of IEEE INFOCOM*, 2002.

[2] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, R. Vijaykumar, , and P. Whiting. Cdma data qos scheduling on the forward link with variable channel conditions. Technical report, Bell Laboratories, Lucent Technologies, 2000.

[3] M. Armony. *Queuing Networks with Interacting Network Resources*. PhD thesis, Stanford University, 1999.

[4] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushayana, , and A. Viterbi. CDMA/HDR: A bandwidth-efficient high-speed wireless data service for nomadic users. *IEEE Communication Magazine*, 38:70–77, July 2000.

[5] H. Chen and D. Yao. *Fundamentals of Queuing Networks*. Springer, 2001.

[6] A. Eryilmaz, R. Srikant, and J. Perkins. Stable scheduling policies for fading wireless channels, 2002.

[7] J. M. Holtzman. CDMA forward link waterfilling protocol. *Vehicular Technology Conference, Spring-2000*, Spring 2000.

[8] S. S. Kulkarni and C. Rosenberg. Opportunisitc scheduling policies for wireless systems with short term fairness constraints. To Appear in Proceedings of IEEE GLOBECOM, December 2003.

[9] K. Kumaran and L. Qian. Uplink scheduling in CDMA packet-data systems. In *Proceedings of IEEE INFOCOM*, 2003.

[10] X. Liu, E. K. P. Chong, and N. B. Shroff. Transmission scheduling for efficient wireless utilization. In *Proceedings of IEEE INFOCOM*, 2000.

[11] X. Liu, E. K. P. Chong, and N. B. Shroff. *A Framework for Opportunistic Scheduling.* PhD thesis, Purdue University, December 2001.

[12] Y. Liu, S. Gruhl, and E. Knightly. WCFQ: An opportunistic wireless scheduler with statistical fairness bounds. *To appear in IEEE Transactions on Wireless Communication*, September 2003.

[13] Y. Liu and E. Knightly. Opportunistic fair scheduling over multiple wireless channels. In *Proceedings of IEEE INFOCOM*, 2003.

[14] S. Lu, V. Bharghavan, and R. Srikant. Fair scheduling in wireless packet networks. *IEEE/ACM Transactions on Networking*, 7:473–489, August 1999.

[15] M. J. Neely, E. Modiano, and C. E. Rohrs. Power and server allocation in a multi-beam satellite with time varying channels. In *Proceedings of IEEE INFOCOM*, 2002.

[16] T. Ng, I. Stoica, and H. Zhang. Packet fair queuing algorithms for wireless networks with location dependent errors. In *Proceedings of IEEE INFOCOM*, 1998.

[17] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly. Opportunistic media access for multirate Ad-Hoc networks. In *Proceedings of ACM MOBICOM*, 2002.

[18] S. Shakkottai and A. Stolyar. Scheduling algorithms for a mixture of real-time and non-real-time data in HDR. In 17^{th} *International Teletraffic Congress*, Septmber 2001.

[19] P. Viswanath, D. N. C. Tse, and V. Anantharam. Asymptotically optimal waterfilling in vector multiple access channels. *IEEE Transctions on Information Theory*, 47, January 2001.

[20] P. Viswanath, D. N. C. Tse, and R. Laroria. Opportunistic beamforming using dumb antennas. *IEEE Transactions on Information Theory*, 48:1277–1294, June 2002.

APPENDIX

A. PROOF OF THEOREM 3

We shall prove the claim that a policy $Kargmax_i(a_i q_i \mu_{ik})$ is throughput optimal for any positive constants a_i. (We have adopted a slight change of notation for convenience. Instead of μ_i^k, we use $\mu_{i,k}$ for the data rate for user i, on interface k.) We use a fluid approach similar to the one used in [2]. The method is *similar* with few differences for taking into account the constraints imposed by the problem structure. Specifically, "a user cannot be assigned more than one interface and an interface cannot be assigned more than one user", will be taken into account in the proof. We start with the following assumptions. The combined service rate process for all user-interface pairs is assumed to be an aperiodic, irreducible, finite, ergodic Markov chain with a unique stationary distribution denoted by $\pi = (\pi^1, \cdots, \pi^{|\mathcal{M}|})$, where the superscript denotes the channel state. We also assume that the combined arrival process forms a similar aperiodic, irreducible, finite, ergodic Markov chain and we denote the mean arrival rate for user i in one timeslot by λ_i. We consider only those scheduling rules which map the system state (which is a combination of the channel state, arrival state, each user's queue state with the associated delays), to every possible $\mathcal{N} \times \mathcal{K}_{\{0,1\}}$ matrix B satisfying the constraints $\sum_i b_{ik} \leq 1$, $\sum_k b_{ik} \leq 1$ with a stationary probability distribution. We also assume that the system state evolves as a aperiodic, irreducible, Discrete Time Markov Chain.

A.1 Fluid Approximation

As in section 5, 6 of [2] define the following random functions associated with the system state process $S(t)$.

- $F_i(t)$ denotes the total amount of service arrivals up to time t for a user i.
- $\hat{F}_i(t)$ denotes the total amount of service departures up to time t for a user i.
- $G^m(t)$ denotes the number of timeslots up to time t the channel is in state m.
- $Q_i(t) = F_i(t) - \hat{F}_i(t)$ denotes the queue size for user i up to time t.
- $U_i(t) = \inf\{s \leq t : F_i(s) > \hat{F}_i(t)\}$ and $W_i(t) = t - U_i(t)$. $W_i(t)$ denotes the waiting time of the oldest unfinished service requirement for user i at time t. $U_i(t)$ denotes the arrival time of the oldest unfinished service requirement at the head of the queue for user i.

We define $\|S(t)\| = \sum_i (Q_i(t) + W_i(t))$. We assume a sequence of such random functions for which the initial condition $\|S(t)\| = \sum_i (Q_i(0) + W_i(0)) = n$. Let $S^n(t)$ denotes such a process. Thus $\|S^n(0)\| = n$. We assume that arrivals happen from time $-n$ onward for process S^n; but the service starts from time 0 only. All the above processes $F_i, \hat{F}_i, G^m, Q_i, U_i, W_i$ are superscripted with n to denote that these are processes with the above defined initial condition. Let $X^n(t)$ be the vector of all these processes. To obtain the fluid approximation (refer to [5]) define the scaled processes for all these processes as follows. $x^n(t) = \frac{1}{n} X^n \lfloor nt \rfloor$. Also extend the definition of processes F_i^n, W_i^n, U^n to $t = [-n, 0]$ to take into account the fact that some arrivals happen before zero time so as to satisfy the initial system condition.

A.2 Theorem on Fluid Approximation

For any sequence of $X^l(t), l \in \{0, 1, 2, \cdots\}$, there exists a further subsequence of processes X^n, $\{n\} \subset \{0, 1, 2, \cdots\}$ such that the following holds $\forall i \in \mathcal{N}, m \in \mathcal{M}$, w.p. 1. There exist nonnegative, nondecreasing, RCLL (right continuous with left limits) functions $f_i \in [-1, \infty)$ and Lipschitz continuous in $[0, \infty)$. Also there exist nondecreasing, nonnegative Lipschitz continuous functions $\hat{f}_i, g^m, \in [0, \infty)$ and Lipschitz continuous nonnegative functions $q_i \in [0, \infty)$.

Because of the ergodic assumption on the structure of the arrival Markov chain following holds. (In the following parts of proof u.o.c. denotes uniform convergence on compact intervals, c.c.p. denotes convergence at continuity points.)

$$\begin{aligned} &for\ t \geq 0, \quad && f_i^n(t) \overset{u.o.c.}{\to} f_i(t); \;\; f_i(t) - f_i(0) = \lambda_i t \\ &for\ t \geq -1, \quad && f_i^n(t) \overset{c.c.p.}{\to} f_i(t); \;\; \sum_i f_i(0) \leq 1 \end{aligned}$$

Because of the ergodic assumption on the channel state Markov chain following holds.

$$for\ t \geq 0,\ \ g^{n,m}(t) \overset{u.o.c.}{\rightarrow} g^m(t);\ \ g^m(t) - g^m(0) = \pi^m t$$

For $[t_1, t_2] \subset [0, \infty)$, $\hat{f}_i^n(t_2) - \hat{f}_i^n(t_1) \leq \mu_{max}(t_2 - t_1 + \frac{2}{n})$ holds trivially. Using this fact and noting the fact that $\hat{f}_i(t)$ are nondecreasing, a further subsequence can be found such that,

$$\begin{aligned} \forall\, t \geq 0, \qquad & \hat{f}_i^n(t) \overset{u.o.c.}{\rightarrow} \hat{f}_i(t) \leq f_i(t);\ \ \hat{f}_i(0) = 0 \\ & q_i^n(t) \overset{u.o.c.}{\rightarrow} q_i(t) = f_i(t) - \hat{f}_i(t) \end{aligned}$$

Hence $t \geq -1$,

$$w_i^n(t) \overset{c.c.p.}{\rightarrow} w_i(t);\ \ u_i^n(t) \overset{c.c.p.}{\rightarrow} u_i(t);\ \ w_i(t) = t - u_i(t)$$

Whenever $q_i(t) > 0$, choose t_1, t_2 s.t. $\forall t \in [t_1, t_2],\ q_i(t) > 0$, which is possible because of the Lipschitz continuity of $q_i(t)$.

Let $0 < \delta < \inf_{t \in [t_1, t_2]} q_i(t)$. As $q_i^k(t)$ converges uniformly to $q_i(t)$, we can choose subsequence $\{j\}$ such that $\exists L,\ \forall l > L,\ \forall t \in [t_1, t_2],\ q_i^l(t) = \frac{1}{l} Q_i^l(jt) > \delta$.

The above statement implies that $\forall t \in [t_1, t_2],\ Q_i^l(lt) > l\delta > \mu_{max}$, as l can be made arbitrarily large. This means that a service is not wasted due to the empty queues.

Now assume that, $\forall i,\ \ q_i(t) \neq 0$ and $\vec{q}(t)$ is such that, $\forall i, j, k, \hat{k}, m,\ a_i \mu_{i,k}^m q_i(t) \neq a_{j,\hat{k}} \mu_j^m q_j(t)$. Then $Kargmax$ is uniquely defined for $\forall m$. Also $\exists \epsilon \geq 0$ such that, if $|\vec{q}(t) - \vec{\tilde{q}}(t)| \leq \epsilon \Rightarrow$ $Kargmax$ is still uniquely defined $\forall m \in \mathcal{M}$ and $Kargmax_{i,k}\{a_i \mu_{ik}^m q_i(t)\} = Kargmax_{i,k}\{a_i \mu_{ik}^m \tilde{q}_i(t)\}$.

Hence for such $\vec{q}(t)$, $\exists t_1, t_2$, s. t., $\forall t \in [t_1, t_2]$, $S_{Q^*}^{ik}(\vec{q}(t), m_t)$ remains constant for a given channel state m, where $S_{Q^*}^{ik}$ is defined to be the service rate offered by policy Q^* to the user i on interface k when the queue size is $\vec{q}(t)$ and the channel state is m. Also the following relationship holds true.

$$\left| \hat{F}_i^l(lt_2) - \hat{F}_i^l(lt_1) - \sum_{t=t_1}^{t=t_2} \sum_k S_{Q^*}^{ik}(\vec{Q}(lt), m_{lt}) \right| \leq 2\mu_{max}$$

By the Strong Law of Large Numbers, for large enough l,

$$\sum_{t=t_1}^{t=t_2} \sum_k S_{Q^*}^{ik}(\vec{Q}(lt), m_t) = l(t_2 - t_1) \sum_{m,k} \pi^m S_{Q^*}^{ik}(\vec{q}(t), m)$$

Hence, we can write the following equation,

$$\hat{f}_i(t_2) - \hat{f}_i(t_1) = (t_2 - t_1) \sum_{m,k} \pi^m S_{Q^*}^{ik}(\vec{q}(t), m) \qquad (8)$$

Also we claim that $\hat{f}_i(t_1) > f_i(0),\ \ t \in [t_1, \infty), \Rightarrow \lambda_i w_i(t) = q_i(t)$. This claim is similar to the Little's theorem and we prove it as follows. As $u_i(t) = \inf_s\{s \leq t : f_i(s) > \hat{f}_i(t)\}$ and $f_i(s) = \lambda_i s + f_i(0) \Rightarrow \inf_s\{s \leq t : \lambda_i s + f_i(0) > \hat{f}_i(t)\} \Rightarrow s = \frac{\hat{f}_i(t) - f_i(0)}{\lambda_i}$. So we get $u_i(t) = t - w_i(t) = \frac{\hat{f}_i(t) - f(0)}{\lambda_i}$. Thus the above equation follows as $f_i(t) = \lambda_i t + f_i(0)$. Note that Lipschitz continuity implies absolute continuity which implies derivative exists w.r.t. Lebesgue measure almost everywhere. In further discussion whenever we take a derivative we assume that the derivative at that point is defined (which is true a.e.). In particular we have,

$$q_i(t) > 0,\ \ \Rightarrow\ \ \frac{d}{dt}\hat{f}_i(t) = \sum_{mk} \pi^m S_{Q*}^{ik}(\vec{q}(t), m)$$

A.3 Quadratic Lyapunov Function

Let $L(y) = \frac{1}{2}\sum_i a_i y_i^2$ for a vector y. Thus, $L(q(t)) = \frac{1}{2}\sum_i a_i q_i(t)^2$. Clearly L is a Lipschitz continuous function w.r.t. t and $L(q(0)) \leq \frac{1}{2}\sum_i a_i$.

We claim that, $L(q(t)) > 0 \Rightarrow \frac{d}{dt}L(q(t)) = L^{'}(q(t)) < 0$. The proof is as follows. Assume $q_i(t) > 0$ for $\forall i$ without loss of generalization. Let Q be any other stationary feasible (randomized) rate stable policy which depends only on the current channel state m. We can prove that for such a policy $\frac{d}{dt}\hat{f}_i(t) = \sum_{mk} \pi^m S_Q^{ik}(\vec{q}(t), m)$. Where, S_Q^i is now understood to be the expected (because Q may be a randomized policy) service rate offered to user i. Hence $\forall i, \lambda_i < \sum_{mk} \pi^m S_Q^{ik}(\vec{q}(t), m)$. Then,

$$\begin{aligned} & L^{'}(q(t)) \\ &= \frac{d}{dt}\frac{1}{2}\sum_i a_i(f_i(t) - \hat{f}_i(t))^2 \\ &= \sum_i a_i q_i(t)(\lambda_i - \hat{f}_i^{'}(t)) \\ &= \sum_i a_i q_i(t)\lambda_i - \sum_{i,m,k} a_i q_i(t)\pi^m S_Q^{ik}\{\vec{q}(t), m\} \\ &\quad + \sum_{i,m,k} a_i q_i(t)\pi^m \Big[S_Q^{ik}\{\vec{q}(t), m\} - S_{Q^*}^{ik}\{\vec{q}(t), m\}\Big] \\ &= \sum_i a_i q_i(t)\Big[\lambda_i - \sum_{m,k} \pi^m S_Q^{ik}\{\vec{q}(t), m\}\Big] \\ &\quad + \sum_m \pi^m \Big[\sum_{i,k} a_i q_i(t) S_Q^{ik}\{\vec{q}(t), m\} - \sum_{i,k} a_i q_i(t) S_{Q^*}^{ik}\{\vec{q}(t), m\}\Big] \end{aligned}$$

The first term in the last equation is strictly negative by the assumption that policy Q is rate stable. Now consider the second term. As the policy Q^* is a $Kargmax$ policy the second term is also bounded above by zero. Thus $L^{'}(q(t))$ is strictly negative which proves the claim. Thus $L(q(t))$ is a decreasing function a.e. and as it is Lipschitz continuous, for any given $\epsilon > 0, \exists\ T\ s.t.\ \forall t \geq T,\ L(q(t)) < \epsilon$. This also implies that for large t, $\sum_i q_i(t) \leq \delta$. Then as λ_i are nonzero by assumption, $\hat{f}_i(t) = f_i(t) - q_i(t) > f_i(0)$, and hence $w_i(t) = q_i(t)/\lambda_i$. Thus for large enough $T, \forall t > T$,

$$\begin{aligned} \sum_i (q_i(t) + w_i(t)) &\leq (1 + \frac{1}{\lambda_{min}})\delta < 1 \\ \Rightarrow \limsup_{n \to \infty} \left\| S^n(nT) \right\| &< 1 \end{aligned}$$

Hence the conditions for applying theorem 4 in [2] are satisfied and this proves the throughput optimality of the $Kargmax$ rule. We also note that, this method can also be used to prove the conjecture at the end of the section 4 in [15]. Specifically we need to prove equation 8 for the power allocation physical constraints. In general *if the constraints placed on the data rate values by the time varying channel conditions and by the other physical limitations are independent of the queue sizes*, then our proof can be applied to show that a $Kargmax_{i,k}(a_i q_i(t)\mu_{i,k}(t))$ type of policy satisfying the physical constraints is throughput optimal.