International Journal of Computing Academic Research (IJCAR)
ISSN 2305-9184 Volume 4, Number 1(February 2015), pp. 1-11

http://www.meacse.org/ijcar

Utility Maximization Scheduling in Multi-channel Wireless Networks

Xiaoling Xiong and Ruixue Liu
School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China

Abstract

This paper considers the problem of designing utility maximization scheduling algorithms for multi-channel(e.g, OFDM-based) wireless down-link systems. We extend Lyapunov Optimization to design a throughput-utility maximizing algorithm that uses a queue-based and delay-based Lyapunov functions, where the delay uses explicit delay information from the head-of-line packet destined to each user. Our approach provably achieves the maximum utility, and empirically outperforms the previous solution.

Keywords: Utility maximization, Scheduling algorithms, Multi-channel.

Introduction

In the OFDM-based wireless down-link systems(e.g., WiMax[1])and LTE[2]), the bandwidth available at the base-station is partitioned into hundreds or thousands of orthogonal frequency bands(or channels). In every timeslot, a given frequency band can be allocated to one and only one user, but a given user can be served by multiple frequency bands simultaneously, and the allocation can change over every time-slot, depending upon the channel quality and the queue backlogs.

It is well known that the MaxWeight algorithm [3]-[4] is throughput-optimal for this system. However, the delay performance of this algorithm is well known to be poor, both from an average and a worst-case delay sense (for instance, see [5]-[6]). The works in [7]-[9] all use queue-based transmission rules to treat joint stability and utility optimization. However, work in [10] introduces a delay-based Lyapunov function for proving stability, where the delay of the head-of-line packet is used as a weight in the max-weight decision. To our knowledge, there are no prior works that combine these two scheduling rules to address the important issue of joint stability and throughput-utility optimization in multi-user multi-channel system. This paper fills this gap. We use a queue-based and delay-based Lyapunov functions to treat this issue via the Lyapunov Optimization technique from the prior work[11]. We evaluate the performance of our approach through simulations, and it show that our scheduling algorithm outperforms the previous solution.

1. System Model

A. Model

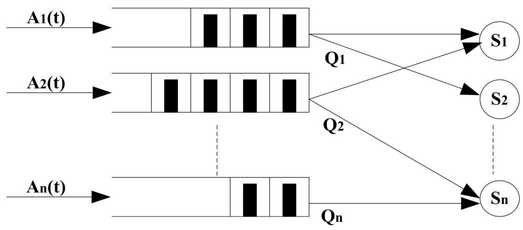

Fig.1. System Model

We consider a multi-user multi-channel system with stochastic connectivities as shown in Figure 1. For simplicity, the number of users is assumed to be equal to the number of channels. There are n users (queues) and n servers(channels). The system is time-slotted. In a time-slot, a server can be allocated to only one queue, but a queue can get service from multiple servers. The connectivity between queues and servers is time-varying, i.e., it can change between "ON" and "OFF" from time to time.

We let Q_i denote the queue(at the BS) associated with the $i-th$ user, where $\boldsymbol{Q}(t)=(Q_1(t),...,Q_n(t))$ denotes the backlogs(in units of packers) of each queue at the end of time-slot t. We assume that each queue has infinite buffer capacity so that no packets are ever dropped. Let S_j denote the $j-th$ server, where $\boldsymbol{S}(t)=(S_1(t),...,S_n(t))$ denotes a server condition vector for slot t, and $S_j(t)\in\{ON,OFF\}$. We assume that $\boldsymbol{S}(t)$ is i.i.d. over slots and known to the BS at the beginning of each slot t. Let $\boldsymbol{A}(t)=(A_1(t),...,A_n(t))$ be the arrival vector, where $A_i(t)$ denote the number of packer arrivals to queue Q_i in time-slot t. For simplicity, we assume all packets have fixed size, and that there is at most one packet arrival to each queue per slot, so that $A_i(t)\in\{0,1\}$ for all queues i and slots t. We assume that $\boldsymbol{A}(t)$ are i.i.d. every time slot with $E\{\boldsymbol{A}(t)\}=\lambda\triangleq(\lambda_1,...,\lambda_n)$. We let $\mu_i(t)$ represents the amount of packets successfully served on slot t. The queuing dynamics as follows:

$$Q_i(t+1)=\max[Q_i(t)-\mu_i(t),0]+A_i(t) \tag{1}$$

As it clear from the above equation, the queues can store any number of packets in the buffers, and the system does not drop any packets.

B. Time Varying Allocation Reliability

Let $\mu_{ij}(t)$ denote the amount of packets in queue i successfully served by serve j on slot t. For simplicity, we assume that each server can server at most one packet per slot, so that $\mu_{ij}(t)\in\{0,1\}$ for all servers and all slot t. Let $\boldsymbol{x}(t)=(x_1(t),...,x_n(t))$ denote a transmission vector, where $x_i(t)\in\{0,1\}$, and $x_i(t)=1$ if queue i attempts transmission on slot t. We assume that $\boldsymbol{x}(t)$ must be drawn from some general allowable transmission set $\mathbb{X}$, i.e., $\boldsymbol{x}(t)\in\mathbb{X}$ for all t. We also assume that each set $\mathbb{X}$ is time invariant and compact. The transmission vector $\boldsymbol{x}(t)$ and server condition vector $\boldsymbol{S}(t)$ jointly determine the probability of successful allocation on each slot. Specifically, given particular $\boldsymbol{x}(t)$ and $\boldsymbol{S}(t)$ vectors, the probability of server j successfully allocated to queue i is given by a reliability function:

$$\Pr[S_j \text{ allocated to } Q_i \mid \boldsymbol{x}(t),\boldsymbol{s}(t)] = \Psi_{ij}(\boldsymbol{x}(t),\boldsymbol{s}(t)) \tag{2}$$

The reliability function $\Psi_{ij}(\boldsymbol{x},\boldsymbol{s})$ is general for all i and $j\in\{1,\dots,n\}$ and is assumed only to take real values between 0 and 1 (representing probabilities), and to have the property that $\Psi_{ij}(\boldsymbol{x},\boldsymbol{s})=1$ whenever $x_i=1$ and $S_j=\mathrm{ON}$, and 0 otherwise. In practice, $\boldsymbol{S}(t)$ represents the result of a channel estimation that is done every slot. The estimation might be inexact, in which case the reliability function $\Psi_{ij}(\boldsymbol{x},\boldsymbol{s})$ represents the probability that the actual network channels on slot t are sufficient to support the attempted transmission of queue i. We assume the reliability function is known.

We assume that ACK/NACK information is given at the end of the slot to inform each queue if its transmission was successful or not. Packets that are not successful do not leave the queue. With this model of allocation success, the serve variable $\mu_{ij}(t)$ is given by:

$$\mu_{ij}(t) = x_i(t)I_{ij}(t) \tag{3}$$

Where $I_{ij}(t)$ is an indicator variable that is 1 if the transmission of queue i is successfully served, and 0 otherwise. That is:

$$I_{ij}(t)=\begin{cases}1 & \text{with probability } \Psi ij(\boldsymbol{x}(t),\boldsymbol{s}(t))\\ 0 & \text{with probability } 1-\Psi ij(\boldsymbol{x}(t),\boldsymbol{s}(t))\end{cases} \tag{4}$$

The successes/failures over each server on slot t are assumed to be independent of past history given the current $\boldsymbol{x}(t)$ and $\boldsymbol{S}(t)$ values.

C. The Optimization Objective

Our goal is to design a network scheduling policy that maximizes a general network utility metric which is a function of the achieved throughput vector. Specifically, let $y_i(t)$ be the amount of $\text{user}-i$ data served in slot t, and define the throughput $\overline{y_i}(t)$ for user i as follows:

$$\overline{y_i} \triangleq \lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}[y_i(\tau)] \tag{5}$$

Then we seek to solve the following utility maximization problem:

Maximize: $g(\overline{\boldsymbol{y}})$ (6)

Subject to: $\overline{\boldsymbol{y}} \in \Lambda$ (7)

$0 \le \overline{y}_i \le \lambda_i$ for all $i \in \{1,...,n\}$ (8)

Let Λ be the network capacity region of the wireless downlink, define as the closure of the set of all achievable throughput vectors $\overline{\boldsymbol{y}} \triangleq (\overline{y}_1,...,\overline{y}_n)$. Where in the above we denote by $g(\bullet)$ a generic utility function that is concave,continuous,non-negative, and non-decreasing.

2. The Optimal Scheduling Algorithm

In this section, we use the framework of Lyapunov Optimization to develop an optimal scheduling algorithm for our model.

A. Constructing Lyapunov Function

Let $H_i(t)$ represent the waiting time of the head-of-line packet in queue i on slot t , and define $H_i(t)$ =0 if there are no packets in queue i at this time. A new packet that arrives to an empty queue on slot t is not placed to the head-of-line until the next slot, and is designated to have a waiting time of 1 at slot $t+1$. Define $\alpha_i(t)$ as an indicator variable that is 1 if $Q_i(t) > 0$, and is zero if the queue is empty. Let $\beta_i(t) = 1 - \alpha_i(t)$. Similar to [10], we observe that $H_i(t)$ satisfies the following update rule:

$$H_i(t+1) = \alpha_i(t)\max[H_i(t) + 1 - \mu_i(t)T_i(t), 0] + \beta_i(t)A_i(t) \tag{9}$$

Where $T_i(t)$ represents the inter-arrival time between the head-of-line packet and the subsequent packet(possibly unknown to the BS if the subsequent packet has not yet arrived). Because arrivals are Bernoulli, $T_i(t)$ is a geometric random variable with success probability λ_i , that takes values in the set {1, 2, 3...}.

The equation (9) can be understood as follows: If $\alpha_i(t) = 0$, then $\beta_i(t) = 1$ so that queue i is empty. In this case, the value of $H_i(t+1)$ is 1 if and only if there is a new arrival on slot t . Alternatively, if $\alpha_i(t) = 1$, then $\beta_i(t) = 0$. Suppose in this case that the head-of-line packet is not served $(\mu_i(t) = 0)$, then its delay increases by 1, as described by (9). On the other hand, if the head-of-line packet is served $(\mu_i(t) = 1)$, then the next packet enters the head-of-line, with a total waiting time equal to $H_i(t)+1-T_i(t)$ (where the additional "+1" comes because this operation takes one more slot). The above dynamics also capture the possibility that the inter-arrival time is greater than or equal to $H_i(t)+1$, in which case the queue is empty on slot $t+1$ with $H_i(t+1)=0$.

Define $\boldsymbol{\Theta}(t) \triangleq [\boldsymbol{Q}(t);\boldsymbol{H}(t)]$, where $\boldsymbol{Q}(t)$ and $\boldsymbol{H}(t)$ are vectors of queues in (1) and the head-of-line values in (9). We use the following non-negative Lyapunov function:

$$L(\Theta(t)) \triangleq \frac{1}{2}\sum_{i=1}^{n} Q_i(t)^2 + \frac{1}{2}\sum_{i=1}^{n} \lambda_i H_i(t)^2 \tag{10}$$

B. Minimizing the Drift-Minus-Utility

Define $\Delta(\Theta(t))$ as the one-step conditional Lyapunov drift:

$$\Delta(\Theta(t)) \triangleq \mathbb{E}\{L(\Theta(t+1)) - L(\Theta(t)) \mid \Theta(t)\} \tag{11}$$

Using our Lyapunov Optimization framework in [1], our strategy is to make transmission decisions to minimize a bound on the following drift-minus-utility expression every slot:

$$\Delta(\Theta(t)) - V\mathbb{E}\{g(y(t)) \mid \Theta(t)\} \tag{12}$$

Where V is a non-negative control parameter that is chosen as desired, and will affect an explicit utility-delay trade-off. We have the following preliminary lemmas:

Lemma 1: Every slot t, for any value of $\Theta(t)$, and under any control policy, the Lyapunov drift satisfies:

$$\Delta(\boldsymbol{\Theta}(t)) \le B + \sum_{i=1}^{n} Q_i(t)\mathbb{E}\{(A_i(t)-\mu_i(t)) \mid \boldsymbol{\Theta}(t)\} + \sum_{i=1}^{n} \lambda_i H_i(t)\mathbb{E}\{1-\mu_i(t)T_i(t) \mid \boldsymbol{\Theta}(t)\} \tag{13}$$

Where B is a finite constant.

Proof: The proof follows by squaring (1) and (9), and using $\alpha_i(t)H_i(t)=H_i(t)$, $\alpha_i(t)\beta_i(t)=0$, and is omitted for brevity.

Lemma 2: Every slot t, for any value of $\Theta(t)$, and under any control policy for which $T_i(t)$ is independent of $\mu_i(t)$ and $\Theta(t)$ for all queue i, we have:

$$\begin{aligned}\Delta(\boldsymbol{\Theta}(t)) - V\mathbb{E}\{g(y(t)) \mid \boldsymbol{\Theta}(t) \le B - V\mathbb{E}\{g(y(t)) \mid \boldsymbol{\Theta}(t)\} + \sum_{i=1}^{n} Q_i(t)\mathbb{E}\{A_i(t) \mid \boldsymbol{\Theta}(t)\} + \\ \sum_{i=1}^{n} H_i(t)\mathbb{E}\{\lambda_i \mid \boldsymbol{\Theta}(t)\} - \sum_{i=1}^{n}(Q_i(t)+H_i(t))\mathbb{E}\{\sum_{i=1}^{n} x_i(t)\Psi_i(\boldsymbol{x}(t),\boldsymbol{S}(t)) | \boldsymbol{\Theta}(t)\}\end{aligned} \tag{14}$$

Proof: Using $\mu_i(t)=\sum_{j=1}^{n}\mu_{ij}(t)$, $\mu_{ij}(t)=x_i(t)I_{ij}(t)$, and define $\chi(t)=[\boldsymbol{S}(t),\Theta(t),\boldsymbol{x}(t)]$, then we have $\mathbb{E}\{I_{ij}(t) \mid \chi(t)\}=\Psi_{ij}(\boldsymbol{x}(t),\boldsymbol{s}(t))$, and let $\varphi(t)=[\Theta(t),\mu_i(t)]$, using the law of iterated expectations, we have:

$$\mathbb{E}\{\mu_i(t)T_i(t) \mid \Theta(t)\} = \mathbb{E}\{\mathbb{E}[\mu_i(t)T_i(t) \mid \varphi(t)] \mid \Theta(t)\} = \mathbb{E}\{\mu_i(t)\mathbb{E}[T_i(t) \mid \varphi(t)] \mid \Theta(t)\} = \frac{1}{\lambda_i}\mathbb{E}\{u_i(t) \mid \Theta(t)\}$$

Where we have used the fact that $T_i(t)$ is independent of $\varphi(t)$ and is a geometric random variable with $\mathbb{E}\{T_i(t)\} = 1/\lambda_i$. Lemma 2 follows by plugging these identity into Lemma 1 and subtracting $V\mathbb{E}\{g(y(t)) \mid \Theta(t)\}$ from both sides.

C. The Optimal Scheduling Algorithm

Our dynamic scheduling policy below makes control decisions for $\boldsymbol{x}(t)$ to minimize the right hand side of the drift-minus-utility bound in Lemma 2. Every slot t, observe $\boldsymbol{Q}(t)$, $\boldsymbol{H}(t)$, and $\boldsymbol{S}(t)$, and perform the following operations:

1) Transmission Scheduling: Observe $\Theta(t)$ and $\boldsymbol{S}(t)$ and choose a transmission vector $\boldsymbol{x}(t)$ to solve the following :

Maximize: $V\mathbb{E}\{g(y(t)) \mid \Theta(t)\} + \sum_{i=1}^{n}(Q_i(t) + H_i(t))\mathbb{E}\{\sum_{j=1}^{n} x_i(t)\Psi_{ij}(\boldsymbol{x}(t), \boldsymbol{s}(t)) \mid \Theta(t)\}$ (15)

Subject to: $\boldsymbol{x}(t) \in \mathbb{X}$ (16)

Where $y(t) = \sum_{i=1}^{n}\mu_i(t) = \sum_{i=1}^{n}\sum_{j=1}^{n}\mu_{ij}(t) = \sum_{i=1}^{n}\sum_{j=1}^{n}x_i(t)I_{ij}(t)$

2) Queue Update: Update the queue using (1) and (9).

3. The Performance Analysis

Theorem 1. Suppose there exists an $\varepsilon \geq 0$ such that $\lambda + 2\varepsilon\mathbf{1} \in \Lambda$, let $\Theta(t) = (\Theta_1(t), ..., \Theta_n(t))$, and assume $\mathbb{E}\{L(\Theta(0))\} < \infty$, then under the scheduling algorithm we proposed, we have:

(a) If $\varepsilon \geq 0$ then all queues $\Theta_i(t)$ are mean rate stable, $i \in \{1, ..., n\}$.

(b) If $\varepsilon > 0$, then all queues are strongly stable and:

$$\lim_{t\to\infty}\sup \frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{i=1}^{n}\mathbb{E}\{|\Theta_i(\tau)|\} \leq \frac{B}{\varepsilon} \quad (17)$$

Where B is same constant from Lemma 1.

Proof: (10) can be written as follows:

$$L(\Theta(t)) = \frac{1}{2}\sum_{i=1}^{n}W_i\Theta_i(t)^2 \quad (18)$$

Where $W_i = \{1, \lambda_i\}$ are a collection of positive weights and (13) can be written as :

$$\Delta(\Theta(\tau)) \leq B - \varepsilon\sum_{i=1}^{n}|\Theta_i(\tau)| \quad (19)$$

Then we first prove part b. Taking expectations of (19) and using the law of iterated expectations yields:

$$\mathbb{E}\{L(\Theta(\tau+1))\} - \mathbb{E}\{L(\Theta(\tau))\} \le B - \varepsilon\sum_{i=1}^{n}\mathbb{E}\{|\Theta_i(\tau)|\} \quad (20)$$

Summing the above over $\tau \in \{0,1,...,t-1\}$ for some slot $t > 0$ and using the law of telescoping sums yields:

$$\mathbb{E}\{L(\Theta(t))\} - \mathbb{E}\{L(\Theta(0))\} \le Bt - \varepsilon\sum_{\tau=0}^{t-1}\sum_{i=1}^{n}\mathbb{E}\{|\Theta_i(\tau)|\} \quad (21)$$

Now assume that $\varepsilon > 0$. Dividing by $t\varepsilon$, rearranging terms, and using the fact that $\mathbb{E}\{L(\Theta(t))\} \ge 0$ yields:

$$\frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{i=1}^{n}\mathbb{E}\{|\Theta_i(\tau)|\} \le \frac{B}{\varepsilon} + \frac{1}{\varepsilon t}\mathbb{E}\{L(\Theta(0))\} \quad (22)$$

The above holds for all slots $t > 0$. Taking a limit as $t \to \infty$ proves part (b).

To prove part (a), we have from (21) that for all slots $t > 0$:

$$\mathbb{E}\{L(\Theta(t))\} - \mathbb{E}\{L(\Theta(0))\} \le Bt \quad (23)$$

Then plugging (18) into (23) yields:

$$\frac{1}{2}\sum_{i=1}^{n}W_i\mathbb{E}\{\Theta i(t)^2\} \le Bt + \mathbb{E}\{L(\Theta(0))\} \quad (24)$$

Therefore, for all $i \in \{1,...,n\}$, we have:

$$\mathbb{E}\{\Theta_i(t)^2\} \le \frac{2Bt}{W_i} + \frac{2}{W_i}\mathbb{E}\{L(\Theta(0))\} \quad (25)$$

However, because $|\Theta_i(t)|$ cannot be negative, we have $\mathbb{E}\{\Theta_i(t)^2\} \ge \mathbb{E}\{|\Theta_i(t)|\}^2$ .Thus for all slots $t > 0$, we have:

$$\mathbb{E}\{|\Theta_i(t)|\} \le \sqrt{\frac{2Bt}{W_i} + \frac{2\mathbb{E}\{L(\Theta(0))\}}{W_i}} \quad (26)$$

Dividing by t and taking a limit as $t \to \infty$ proves that:

$$\lim_{t\to\infty}\frac{\mathbb{E}\{|\Theta_i(t)|\}}{t} \le \sqrt{\frac{2B}{tW_i} + \frac{2\mathbb{E}\{L(\Theta(0))\}}{t^2W_i}} = 0 \quad (27)$$

Thus, all queues $\Theta_i(t)$ are mean rate stable, proving part (a).

Theorem 2. Suppose all queues are initially empty, and let $\boldsymbol{y}^*$ be the optimal time average throughput vector, so that $g(\boldsymbol{y}^*) = g^*$, which is the optimal utility, and $\mu^*(t)$ is the optimal service rate, and we assume that $\mathbb{E}\{\mu^*(t)\} = \mathbb{E}\{A_i(t)\} = \boldsymbol{y}^*$,and the control parameter $V > 0$, then under our scheduling algorithm we have:

$$\liminf_{t\to\infty} g(\overline{\boldsymbol{y}}) \ge g^* - \frac{B}{V} \quad (28)$$

Where $\overline{\boldsymbol{y}} \triangleq (\overline{y}_1, ..., \overline{y}_n)$, and $\overline{y}_i$ is define in (5), and B is same constant from Lemma 1.

Proof: From (14) we have:

$$\Delta(\boldsymbol{\Theta}(t)) - V\mathbb{E}\{g(y(t)) \mid \boldsymbol{\Theta}(t)\} \le B - V\mathbb{E}\{g(y(t)) \mid \boldsymbol{\Theta}(t)\} + \sum_{i=1}^{n} Q_i(t)\mathbb{E}\{(A_i(t) - \mu_i(t)) \mid \boldsymbol{\Theta}(t)\} + \sum_{i=1}^{n} H_i(t)\mathbb{E}\{\lambda_i - \mu_i(t) \mid \boldsymbol{\Theta}(t)\} \tag{29}$$

Then using $g(\boldsymbol{y}^*) = g^*$ and $\mathbb{E}\{\mu^*(t)\} = \mathbb{E}\{A_i(t)\} = y^*$, we have:

$$\Delta(\boldsymbol{\Theta}(t)) - V\mathbb{E}\{g(y(t)) \mid \boldsymbol{\Theta}(t)\} \le B - V\mathbb{E}\{g(y^*(t)) \mid \boldsymbol{\Theta}(t)\} + \sum_{i=1}^{n} Q_i(t)\mathbb{E}\{(A_i(t) - \mu_i^*(t)) \mid \boldsymbol{\Theta}(t)\} + \sum_{i=1}^{n} H_i(t)\mathbb{E}\{\lambda_i - \mu_i^*(t) \mid \boldsymbol{\Theta}(t)\} \tag{30}$$

Taking expectations of (30) yields:

$$\mathbb{E}\{L(\boldsymbol{\Theta}(t+1))\} - \mathbb{E}\{L(\boldsymbol{\Theta}(t))\} - V\mathbb{E}\{g(y(t))\} \le B - V\mathbb{E}\{g(y^*(t))\} + \sum_{i=1}^{n} \mathbb{E}\{Q_i(t)\}\mathbb{E}\{A_i(t) - \mu_i^*(t)\} + \sum_{i=1}^{n} \mathbb{E}\{H_i(t)\}\mathbb{E}\{\lambda_i - \mu_i^*(t)\} \le B - Vg^* \tag{31}$$

The above holds for all $t \ge 0$. Summing over $\tau \in \{0,1,...,t\text{-}1\}$ and dividing by t yields:

$$\frac{1}{t}\{\mathbb{E}\{L(\Theta(t))\} - \mathbb{E}\{L(\Theta(0))\}\} - \frac{V}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\{g(y(\tau))\} \le B - Vg^* \tag{32}$$

Using the fact that $L(\bullet) \ge 0$ and rearranging terms yields:

$$\frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\{g(y(\tau))\} \ge g^* - \frac{B}{V} - \frac{1}{Vt}\mathbb{E}\{L(\boldsymbol{\Theta}(0))\} \tag{33}$$

Using Jensen's inequality in the concave function $g(\bullet)$ yields:

$$g(\overline{y}(t)) \ge g^* - \frac{B}{V} - \frac{1}{Vt}\mathbb{E}\{L(\Theta(0))\} \tag{34}$$

Taking limits of the above as $t \to \infty$ yields (28), this completes the proof.

4. Simulations

In this section, we present numerical results to evaluate the algorithm performance. We consider a 20 users wireless down-link with ON/OFF channels. The arrivals $A_i(t)$ are independent Bernoulli processes, i.i.d. over slots with rates $\lambda_i \in \{0.1, 0.9\}$, $i \in \{1, 2, ..., 20\}$, and the channel state processes $S_j(t)$ are independent and i.i.d. Over slots with $\Pr[S_j(t) = \text{ON}] = 0.5$, $j \in \{1,2,...,20\}$ Every slot, the network controller observes the channel states $S(t) = (S_1(t),...,S_{20}(t))$ and chooses a single queue to serve, transmitting exactly one packet over a served channel that is ON, and no packets over a channel that is not served or that is OFF.

We first fix $V = 1000$ and simulate the queue and delay-based algorithm of Section 2 with t from 0 to 500 time slots, which uses knowledge of the arrival rates $(\lambda_1,...,\lambda_{20})$ .The utility function of achieved throughput $y = (y_1,..., y_{20})$ is $g(y) = \log(1+\sum_{i=1}^{n} y_i)$, Where $\log(\bullet)$ denotes the natural logarithm. The numerical results are shown in Figure 2. It shows the average arrival rate and the average achieved throughput versus time t , we can see the average achieved throughput is almost the same as the arrival rates.

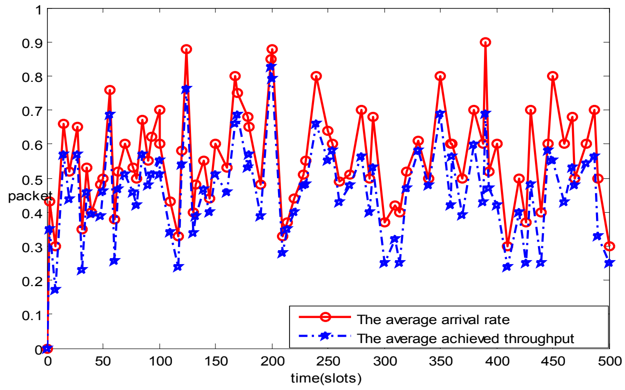

Fig.2.Performance for the queue-based and delay-based algorithm($V = 1000$)

We next consider the throughput and delay as a function of V . We fix the average arrival rate to be 0.5 and vary V from 0 to 1000. The numerical results are shown in Figure 3 and Figure 4.

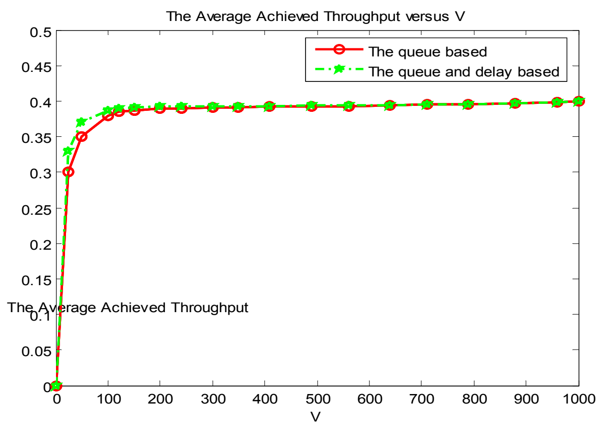

Fig.3. The average achieved throughput versus V

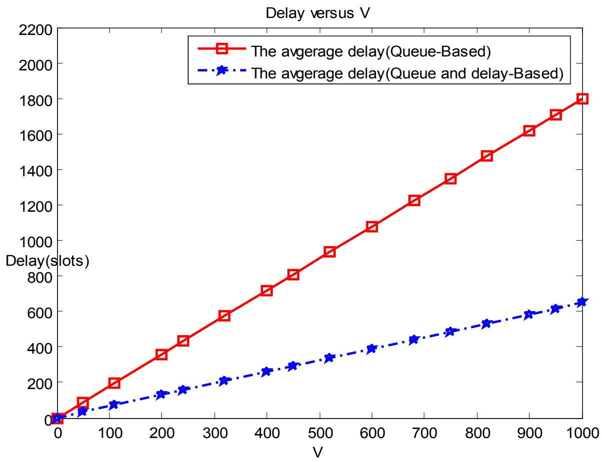

Fig.4. The average delay versus V

Figure 3 shows the resulting average achieved throughput versus V for the queue and delay based with known arrival rates and the queue based algorithm. The average achieved throughput for these two algorithms are very close and converge to the optimal values 0.4 as V is increased. Figure 4 shows the average delay for these two algorithms, the results indicate that the average delays for the queue based algorithm are significantly larger. Thus, our new queue and delay based approach can significantly reduces average delay as compared to the queue based approach.

5. Conclusion

We considered the problem of designing scheduling algorithms for OFDM-based wireless down-link from the point of view of maximizing the throughput-utility of network. We have established a queue-based and delay-based scheduling policy for joint stability and utility optimization. The Lyapunov optimization approach for this queue-based and delay-based problem is significantly different from that of queue-based policies. We believe this policy add significantly to our understanding of scheduling laws in multi-user multi-channel system.

References

[1]W. Forum, Mobile WiMAX Part I: A technical overview and performance evaluation, March 2006, white Paper.

[2]3GPP TR 25.913. Requirements for Evolved UTRA (E-UTRA) and Evolved UTRAN (E-UTRAN). March 2006.

[3]L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. IEEE Trans. Automat. Contr., 4:1936-1948,December 1992.

[4]L. Tassiulas and A. Ephremides, Dynamic server allocation to parallel queues with randomly varying connectivity, IEEE Trans. Inf. Theory,vol. 39, no. 2, pp. 466–478, Mar. 1993.

[5]S. Bodas, S. Shakkotai, L. Ying, and R. Srikant, Scheduling in Multi-Channel Wireless Networks: Rate Function Optimality in the Small Buffer Regime, in Proceedings of ACM SIGMETRICS, 2009.

[6]S. Bodas, S. Shakkottai, L. Ying, and R. Srikant. Low-complexity Scheduling Algorithms for Multi-channel Downlink Wireless Networks. In Proc. IEEE lrifocom, Mar. 2010.

[7]M. J. Neely, E. Modiano, and C. E Rohrs. Dynamic power allocation and routing for time varying wireless networks. IEEE Journal on Selected Areas in Communications, vol. 23, no. 1, pp. 89-103, January 2005.

[8]M. Kobayashi, G. Caire, and D. Gesbert. Impact of multiple transmit antennas in a queued SDMA/TDMA downlink. In Proc. of 6th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC), June 2005.

[9]M. J. Neely and R. Urgaonkar. Optimal backpressure routing in wireless networks with multi-receiver diversity. Ad Hoc Networks (Elsevier), vol.7, no. 5, pp. 862-881, July 2009.

[10]A. Mekkittikul and N. McKeown. A starvation free algorithm for achieving 100% throughput in an input-queued switch. Proc. ICCN, pp. 226-231, 1996.

[11]Michael J. Neely, Stochastic Network Optimization with Application to Communication and Queueing Systems, Oct.2010,pp.1-212