Opportunistic Scheduling of Wireless Links

Vinod Sharma[1], D.K. Prasad[1], and Eitan Altman[2,*]

[1] Dept Elect. Comm. Engg., Indian Istitute of Science,
Bangalore, 560012, India
{vinod, dkp}@ece.iisc.ernet.in
[2] INRIA B.O.93, 2004 Route des Lucioles, 06902
Sophia-Antipolis Cedex, France
Eitman.Altman@sophia.inria.fr

Abstract. We consider the problem of scheduling of a wireless channel (server) to several queues. Each queue has its own link (transmission) rate. The link rate of a queue can vary randomly from slot to slot. The queue lengths and channel states of all users are known at the beginning of each slot. We show the existence of an optimal policy that minimizes the long term (discounted) average sum of queue lengths. The optimal policy, in general needs to be computed numerically. Then we identify a greedy (one step optimal) policy, MAX-TRANS which is easy to implement and does not require the channel and traffic statistics. The cost of this policy is close to optimal and better than other well-known policies (when stable) although it is not throughput optimal for asymmetric systems. We (approximately) identify its stability region and obtain approximations for its mean queue lengths and mean delays. We also modify this policy to make it throughput optimal while retaining good performance.

Keywords: Wireless channel, opportunistic scheduling, greedy scheduling, throughput optimal scheduling, performance analysis.

1 Introduction

We consider the problem of scheduling of several users on a wireless link shared by them. This problem is of relevance in scheduling in uplink and downlink of cellular systems as well as multihop wireless networks. Such problems have been addressed in wireline networks also (e.g., in a router, LAN, switches etc) but in the wireless scenario an added complexity is that the link rates seen by different queues can be different and usually vary randomly in time. Thus the wireline solutions, TDMA, weighted round-robin (when queue lengths and packet sizes may not be available) or strict priority rules such as the $c\mu$ rule ([21]) do not provide reasonable performance in the wireless systems. For good performance one needs to exploit the multiuser rate diversity and use opportunistic scheduling.

---

* The work of this author was supported by a contract with France Telecom.

L. Mason, T. Drwiega, and J. Yan (Eds.): ITC 2007, LNCS 4516, pp. 1120–1134, 2007.

In principle one may allow simultaneous transmission by different users. However, in this paper we will limit ourselves to the case where at a time only one user can transmit.

In [10] it is shown that in a slotted wireless system allocating the slot to the user with the best channel will provide maximum throughput if all the users have always enough data to send. But, by this scheme the throughput received by different users can be very unfair and the mean delays obtained can be quite large ([4]). Thus, if all users always have data to transmit then different approaches have been followed in [3], [5], [12]. When we remove the constraint that all users always have data to transmit, then [2], [6], [17], provide policies which are throughput optimal, i.e., these policies will stabilize the system if any feasible policy will. Throughput optimal policies when the link rates are 0 or 1 were earlier obtained in [19]. Generalization of these results when the arrival processes and the channel availability processes satisfy some burstiness constraints is provided in [20]. Two recent excellent survey-cum-tutorial papers on this topic are [8] and [13]. When we do not consider a slotted system then optimal scheduling policies are also obtained in [7] and [21] Chapter X. When the link rates are constant then the cμ rule is known to be optimal in many different scenarios ([21], [14]).

In this paper we consider a slotted single hop wireless system. The schedular knows the queue lengths and the channel states of each of the queues. The packets arrive at each queue as sequences of independent, identically distributed (*iid*) random variables. The channel rates of each queue also form *iid* sequences. In this scenario, as mentioned above, [2], [6], [17] provide scheduling policies which are throughput optimal. However, often mean delays are of concern. Although the policies provided in [17] also minimize the mean delays under heavy traffic, the policies in [6] provide less mean delays than those provided in [17] when it is not a heavy traffic scenario. Optimal policies which minimize mean delays or queue lengths under heavy traffic have also been studied in [1] when the link rates can be 0 or 1. In this paper we look for policies which minimize mean delay at different operating points. We start with considering the problem of minimizing mean weighted delay. We first show the existence of an optimal policy. This policy is also throughput optimal. However it is obtained numerically and requires the knowledge of the statistics of the arrival traffic and the link rates. Next we look for good sub-optimal policies. MAX-TRANS, also considered in [18], is a one-step optimal greedy policy. This does not require knowledge of channel and traffic statistics and is easy to implement. We compare its performance (discounted average and mean queuelength) to the optimal policy and the policies in [6], [17] and a generalization of the policy in [19]. The MAX-TRANS performs better than the policies in [6], [17] and [19] and is also close to the optimal. However, we will show that it is not throughput optimal for asymmetric traffic (thus there will be traffic rates when the mean delays for MAX-TRANS will be infinite but for the policies in [6] and [17] finite. Even for such cases we have observed that MAX-TRANS provides better performance for the discounted cost problem). We will obtain its (approximate) stability region. We will also provide formulae for approximate mean delay and mean queue length of this policy. Formulae for mean delay and mean queue lengths of policies in [6], [17] and [19] are not available. We will also modify this policy to make it throughput optimal while retaining its performance.

Greedy policies, like MAX-TRANS, have also been considered before (see [8] for comments) and have *not* been recommended because they are not throughput optimal. However, our study indicates that such policies can indeed be useful from performance (e.g., mean delay) point of view and can often be modified to obtain throughput optimal policies.

The paper is organized as follows. In Section 2 we formulate the problem as a Markov Decision Problem and show the existence of an optimal policy. In Section 3 we identify a one-step optimal policy and compare this policy to the optimal policy and other policies available in literature. We find that its performance is quite good as compared to other policies and is also close to the optimal.Thus in Section 4 we study its performance theoretically: we find approximately its stability region and its mean queue length and delays. We verify the accuracy of our approximations with simulations in Section 5. Section 6 concludes the paper.

2 Problem Formulation and Existence of an Optimal Policy

We consider the problem of scheduling transmission of $N \geq 2$ data users (flows) sharing the same wireless channel (server). The system is slotted and multiple transmissions in a slot are allowed. Each user has an infinite buffer to store the data. At the time of transmission, the packets can be arbitrarily fragmented for efficient transmission. We ignore the fragmentation overhead. These assumptions have also been made in [6], [8], [12]. Thus the buffer contents can be considered at the bit level (i.e., queue lengths will be the number of bits in the queue). One of the links is to be scheduled in a slot depending on the current queue lengths and link rates of different users. We denote the queue size of the i^{th} queue at the beginning of the time slot k by $q_k(i)$, the number of arrivals (bits) to queue i in slot k by $X_k(i)$, and the amount of service offered to queue i in slot k by $r_k(i)$. We assume that these parameters can only take non-negative integer values (not really needed). The evolution of the size of the i^{th} queue is given by

$$q_{k+1}(i) = (q_k(i) + X_k(i) - y_k(i)r_k(i))^+, \quad i = 1, ..., N \tag{1}$$

where $(y)^+ = \max(0, y)$ and $y_k(i) = 1$ if i^{th} queue is scheduled in k^{th} slot; otherwise 0.

We assume that the channels of different users can be in any one of the M states in a given slot, where $M < \infty$. The channel state is assumed to be fixed within a slot, but may vary from slot to slot and hence the model captures the time-varying characteristics of a fading channel. The channel rate processes $\{r_k(i), k \geq 1\}$ and the arrival processes $\{X_k(i), k \geq 1\}$ are assumed to be *iid* sequences. Also, these sequences are assumed to be independent of each other. Some of these assumptions may not be true in practical systems but we think this setup captures the essential elements of the general problem and the solutions proposed and the conditions presented should be relevant in the general setup.

We consider the problem of scheduling the channel such that

$$\limsup_{n\to\infty} \sum_{k=1}^{n}\sum_{i=1}^{N} w(i)q_k(i)\alpha^n \tag{2}$$

is minimized where $w(i) > 0$ are weights which reflect the priorities of different users and $0 < \alpha < 1$. We also consider the optimization of the average cost

$$\limsup_{n\to\infty} \frac{1}{n}\sum_{k=1}^{n}\sum_{i=1}^{N} w(i)q_k(i). \tag{3}$$

A policy optimizing (2) will be called an α-discounted optimal policy and a policy optimizing (3) will be called an average cost optimal. By Little's law a policy optimizing (3) also optimizes weighted sums of mean stationary delays. In addition it is throughput optimal. If the Quality of Service (QoS) requirement demands more than mean delays, i.e., has soft or hard delay constraints then (2) can be relevant.

Next we prove the existence of optimal policies for (2) and (3). Obtaining the optimal policies requires information about the statistics of the traffic and link rates and needs to be numerically computed. Therefore, in Section 3 we identify a policy which has performance close to the optimal cost and is better than other well known policies. In Section 4 we study the performance of this policy.

Existence of Optimal Policy

We assume that the rate process $r_k = (r_k(1), ..., r_k(N))$ is component-wise upper bounded by $\bar{r}$, i.e., $\bar{r}(i)$ is the largest value $r_k(i)$ can take, $i = 1, ..., N$. We use the notation $q_k = (q_k(1), ..., q_k(N))$ and $X_k = (X_k(1), ..., X_k(N))$. Also, $J(q, r)$ and $J_\alpha(q, r)$ denote the optimal average cost and α-discounted optimal cost where (q, r) denotes the queue and link rates at time $k = 1$. In addition $J_{\alpha,n}(q, r)$ and $J_n(q, r)$ will denote the n-step optimal costs.

Proof of the following theorem is available in [16](see also [8]).

Theorem 1. Under our assumptions, there exists an α-discounted optimal policy for any α, $0 < \alpha < 1$ and $J_{\alpha,n}(q, r) \to J_\alpha(q, r)$ as $n \to \infty$. If there exists a stationary policy under which the system can reach state $(0, \bar{r})$ in a finite mean time starting from any initial state (q, r) then there also exists an optimal average cost policy.

Under the above conditions, value iteration and policy iteration algorithms for the discounted problem also converge ([9]). Furthermore, for any $\alpha_k \to 1$, $J^* = J(q, r) = \lim_{k\to\infty} J_{\alpha_k}(q, r)$ ([9]).

The condition in the above theorem that there is a stationary policy under which the system can reach state $(0,\bar{r})$ in a finite mean time is implied by the condition that there is a stationary policy under which the process $\{q_k, r_k\}$ is an ergodic Markov chain. This is obviously a necessary condition to have a stable stationary ergodic optimal policy.

3 MAX-TRANS: A Good Suboptimal Policy

Theorem 1 provides the existence of α-discount and average cost optimal policies. We have also seen above that the optimal policy can be obtained numerically via value or policy iteration. However, numerical computations can be cumbersome and may not be feasible in real time for a large number of queues (possible in practical scenarios). Neither does it provide any insight into the problem nor into the optimal policies. Thus in the following we consider suboptimal policies which may be easier to implement and still provide good performance. These policies have been taken from the literature and have been found to have many desirable features. In particular they do not require the statistics of the input traffic and link rates. Also, several of them are throughput optimal ie., they will stabilize a system if it is possible to do so by any feasible policy. We compare the performance of these policies to identify a good policy.

In the following z_k will denote the index of the queue selected by a policy in slot k. Often more than one index will be picked by the criterion used by a policy. Then one can pick one of the selected queues probabilistically.

1. *Maximum Transmission Scheme (MAX-TRANS)*:

$$z_k \stackrel{\triangle}{=} \arg\max w(i) \left(\min(r_k(i), q_k(i) + X_k(i))\right).$$

This policy was used in [18] and compared with several other policies. It was found to provide good performance even when used for a whole frame and in a multihop environment. This is a 'greedy' policy and can be shown to be 1-step optimal for (2) and (3). However we will show that it is not throughput optimal.
2. *Modified Longer Queue First*:

$$z_k = \arg\max \left(w(i) q_k(i).1\{r_k(i) > 0\}\right).$$

For $r(i) \in \{0, 1\}$ this policy is known to be throughput optimal ([19]). For a symmetric system it also minimizes the mean delay for such $r(i)$.
3. *Eryilmaz, Srikant and Perkins Policy [6]*:

$$z_k \stackrel{\triangle}{=} \arg\max w(i) r_k(i) q_k(i).$$

This policy is throughput optimal.
4. *Shakkottai and Stolyar Policy [17]* : Define $\bar{Q}_k = \frac{1}{N}\sum_{i=1}^{N} a_i q_k(i)$ with $a_i = 2$. Then

$$z_k = \arg\max \left\{ w(i)\ r_k(i)\ \exp\left(\frac{a_i q_k(i) - \bar{Q}_k}{1 + \sqrt{\bar{Q}_k}}\right)\right\}.$$

This policy is also throughput optimal and provides minimum delays under heavy traffic.

We compare the performance of these policies for the discounted cost (2) and the average cost (3) for various queueing systems. In the following we provide one example. We also obtain the optimal policies for (2) and (3) via Policy and Value iteration. In addition the costs of the above four policies are obtained via value iteration as well as via simulations.

We consider three queues and a single server. First we consider the discounted (discount factor = 0.94) case and then the undiscounted case. All $w(i)$ in (2) and (3) are taken as 1. Arrival and service distributions for different queues are given in Table 1. We have considered a buffer size of 4 (to keep the complexity of computations small), for numerical computation of the optimal policies and for computing the costs of the above suboptimal policies via value iteration. Thus there are 3375 possible states. We refer to state (0, 0, 0, 0, 0, 0) as state number 1 and state (4, 4, 4, 2, 3, 4) as 3375. The discounted costs are provided in Table 2 for a few initial states. Table 3 has the discounted costs obtained via simulations. Now we consider the queues with infinite buffer. We ran the simulations for 150 slots, two thousand times and obtained their average.

From Tables 2 and 3 we conclude that MAX-TRANS (Policy 1) is the best among all suboptimal policies and is quite close to the optimal for all initial values (although we have shown in these tables the cost for a few initial states due to lack of space, the conclusions hold for other states also). Next is Policy 3. The other two policies can have much worse values. Policy 2 is the worst.

Table 4 contains the results for the average cost problem: via numerical computations and simulations for a buffer size of 4 and via simulations for a buffer size of infinite. Each simulation was run for 2 million slots. Here also we observe the same trend as for the discounted case.

Table 1. Parameters for an example to compare

Q	Link distributions	Input distributions
	2, w.p. = 0.6	2, w.p. = 0.250
1	1, w.p. = 0.3	1, w.p. = 0.168164
	0, w.p. = 0.1	0, w.p. = 0.581836
	3, w.p. = 0.2	3, w.p. = 0.15
2	1, w.p. = 0.5	1, w.p. = 0.151524
	0, w.p. = 0.3	0, w.p. = 0.698476
	4, w.p. = 0.05	3, w.p. = 0.15
3	2, w.p. = 0.6	2, w.p. = 0.20228
	0, w.p. = 0.35	0, w.p. = 0.64772

Table 2. Comparison of policies for discount 0.94; analytical results for buffer size 4

state no.	Optimal	Policy 1	Policy 2	Policy 3	Policy 4
500	79.061	82.275	92.608	83.012	83.965
750	66.469	69.574	76.331	70.076	70.326
1000	82.021	85.221	94.659	85.906	87.013
1250	76.846	80.195	87.415	80.598	81.301
1500	71.546	74.551	83.096	76.078	76.731
2000	93.088	96.026	107.080	97.430	99.213
2250	78.951	81.984	90.004	82.748	83.571
2500	82.918	85.948	94.107	86.597	87.744
3000	82.977	86.120	95.151	86.874	88.171
3250	88.761	91.673	101.870	92.615	93.841

We simulated a large number of different queueing systems and observed the same trend. Actually we observed that MAX-TRANS was always the best among the four suboptimal policies for the discounted cost problem but sometimes the policy 3 was slightly better than policy 1 for the average cost problem (of course there were cases when MAX-TRANS was unstable but Policy 3 was stable). The other two policies were generally worse than policies 1 and 3.

We present one more example to show that the improvement provided by MAX-TRANS can be significant even with respect to the policy 3 which in the above example is always quite close in performance to MAX-TRANS. Let $N = 2$,

Table 3. Comparison for discount 0.94 (simulation with buffer = infinity)

state no.	Policy 1	Policy 2	Policy 3	Policy 4
500	142.73	179.25	145.83	157.90
750	117.60	146.09	120.09	130.20
1000	133.35	167.83	136.57	148.19
1250	149.83	185.79	152.43	164.98
1500	119.76	149.99	122.54	133.12
2000	179.89	228.12	184.98	202.10
2250	154.21	192.40	157.12	170.40
2500	158.88	205.27	163.12	177.17
3000	152.84	191.26	156.11	169.92
3250	160.72	207.31	165.65	180.26

Table 4. Comparison for undiscounted case

Policy	Theoretical cost	Sim cost	Sim with infinite buffer
Optimal	4.4489	4.0587	—
Policy1	4.6770	4.1764	106.5219
Policy2	5.3521	5.2444	$3.3814 * 10^5$
Policy3	4.7452	4.3857	108.9936
Policy4	4.8190	4.6256	$3.5326 * 10^3$

$r_k(1) \equiv 1$, $r_k(2) \equiv 2$ and $P[X_k(1) = 1] = 0.58$ and 0 otherwise, $P[X_k(2) = 2] = 0.4$ and 0 otherwise. Then via simulations we found the average cost of the four policies is 10.820, 14.246, 12.691, 13.863. Thus MAX-TRANS provides 17.3% lower cost than policy 3.

As mentioned above one drawback of MAX-TRANS is that it may not be throughput optimal when the system is not symmetric. We show it by an example. Consider an example with $N = 2$. Let $P[r_k(1) = 1] = 0.01 = 1 - P[r(1) = 0]$ and $P[r_k(2) = 2] = 1$. Also let $P[X_k(1) = 1] = 0.009 = 1 - P[X_k(1) = 0]$ and $P[X_k(2) = 2] = 0.9 = 1 - P[X_k(2) = 0]$. For this example, MAX-TRANS will always serve queue 2 whenever it is non-empty. This happens with probability 0.9. Thus queue 1 gets service at most for probability 0.1. This will make queue 1 unstable. As against this consider a policy that serves queue 1 whenever $r_k(1)$=1. Otherwise serve queue 2. This makes both the queues stable. Thus MAX-TRANS is not throughput optimal.

We will show in Section 4 that for symmetric traffic and link statistics MAX-TRANS is throughput optimal. For asymmetric statistics, as the above example shows, a user with bad channel can get starved for unusually long times making it an unstable queue. Throughput optimal policies do not starve any queue for very long time. Thus we can make MAX-TRANS into a throughput optimal policy, by modifying it appropriately. For example, if we make the following policy: select an appropriately large constant $L > 0$. If all queue lengths are less than L then use MAX-TRANS for scheduling. If one or more queues is larger than L then use policy 3 on those queues to select a queue to serve. If L is very large, the performance of this algorithm will be close to MAX-TRANS. If L is taken small, its performance will be close to policy 3. But for any $L > 0$, the policy will be throughput optimal.

We apply this modified MAX-TRANS on the above example where the MAX-TRANS is unstable with $L = 5$. Then the average cost of this policy and that of policies 2, 3, 4 is 10.70, 8.61, 9.66, 8.81. Thus one sees that the modified MAX-TRANS is stable.

We propose that if the system statistics are symmetric then use MAX-TRANS. Otherwise, if the system is well within the stability region of the MAX-TRANS, then just use this policy. If it is close to the stability region of

MAX-TRANS or outside it but within the stability region of the throughput optimal policies then we can use the modified policy. If the traffic statistics are not known but are asymmetric, then one can use the modified policy with somewhat large L.

Of course to be able to use the above scheme, we need to know the stability region of MAX-TRANS policy. We provide this in the next section. We show that for symmetric conditions MAX-TRANS is throughput optimal. We will also study the performance (mean delays and queue lengths) of this policy. When L is large, then the performance of the modified policy will be close to this. This encourages us to study the MAX-TRANS scheme in more detail.

4 Performance Analysis of MAX-TRANS Policy

As we observed in the last section MAX-TRANS is a promising policy. It is a greedy and one-step optimal policy and compares well with other well-known policies. Thus, it is useful to study its performance in more detail. In this section we obtain the stability region of this policy. We first show that this policy may have a smaller stability region for asymmetric traffic than policies 3 and 4 which are throughput optimal. For simplicity we will first consider this policy for two users and then generalize to multiple users. After studying the stability region of this policy, we obtain approximations for mean queue lengths and delays. We will show via simulations that our approximations work well atleast under heavy traffic. Of course from practical point of view this is the most important region of operation because QoS violation may happen in this region only.

First we provide sufficient conditions for stability for $N = 2$. We will show later on, that for symmetric inter-arrival and link rate distributions these conditions indeed are also necessary. The stability region provided by this theorem is suboptimal for asymmetric traffic. Later we will improve upon these conditions and provide approximate stability region for the asymmetric scenario.

In MAX-TRANS policy if $\min(q_k(1)+X_k(1), r_k(1)) = \min(q_k(2)+X_k(2), r_k(2))$ then we will assign k^{th} slot to queue 1 with probability p, $0 \le p \le 1$ and to queue 2 with probability $(1-p)$. Since $X(i)$, $r(i)$ will often be discrete valued, this event can happen with nonzero probability.
Let $\bar{r} < \infty$ be the largest value $r_k(1)$ and $r_k(2)$ can take.

Theorem 2. If

$$\mathbb{E}[X(1)] < \mathbb{E}[r(1)1_{\{r(1)>r(2)\}}] + p\mathbb{E}[r(1)1_{\{r(1)=r(2)\}}], \tag{4}$$

$$\mathbb{E}[X(2)] < \mathbb{E}[r(2)1_{\{r(2)>r(1)\}}] + (1-p)\mathbb{E}[r(2)1_{\{r(2)=r(1)\}}] \tag{5}$$

then $\{(q_k(1), q_k(2))\}$ stays bounded with probability 1 for any initial condition. Also the inter visit time to set $A = \{x : x \le \bar{r}\}$ by $\{q_k(i)\}$ has finite mean, for $i = 1, 2$.

Proof. We consider another system $\{(\bar{q}_k(1), \bar{q}_k(2))\}$ where the $\bar{q}_{k+1} \triangleq (\bar{q}_{k+1}(1), \bar{q}_{k+1}(2))$ behaves as q_k as long as $\bar{q}_k(1) > \bar{r}$ and $\bar{q}_k(2) > \bar{r}$. If $\bar{q}_k(i) \leq \bar{r}$ then the i^{th} queue is not served in the k^{th} slot. Thus

$$\bar{q}_{k+1}(1) = (\bar{q}_k(1) + X_k(1) - z_k(1))^+$$

where $z_k(1) = 0,$ if $\bar{q}_k(1) \leq \bar{r}$ and if $\bar{q}_k(1) > \bar{r}$, then

$$\begin{aligned} z_k(1) &= r_k(1), \quad \text{if } \ r_k(1) > r_k(2), \\ &= r_k(1), \quad \text{w. p. } \ p \ \text{ if } \ \ r_k(1) = r_k(2), \\ &= 0, \quad \text{otherwise.} \end{aligned} \tag{6}$$

The dynamics of $\bar{q}_k(2)$ behave in the same way. Observe that if $q_k(i) > \bar{r}$, it gets the service as in (6) but in addition it can get service even when $z_k(i) = 0$. Unlike $\{q_k(i), k \geq 0\}$, $\{\bar{q}_k(i), k \geq 0\}$, $i = 1, 2$ is a Markov chain.

Consider the Markov chain $\{(\bar{q}_k(1), \bar{q}_k(2)), k \geq 0\}$. Let $B = \{(x, y) : x \geq x', y \geq y'\}$ for some (x', y'). Then one can easily show that

$$\begin{aligned} &P[(\bar{q}_1(1), \bar{q}_1(2)) \in B | (\bar{q}_0(1), \bar{q}_0(2)) = (q_1, q_2)] \\ \geq\ &P[(q_1(1), q_1(2)) \in B | (q_0(1), q_0(2)) = (q_1, q_2)] \end{aligned}$$

for any (x', y') and (q_1, q_2). We can also construct $\{(\bar{q}_k(1), \bar{q}_k(2), q_k(1), q_k(2)), k \geq 1\}$ on a common probability space ([15]) such that starting from any initial conditions (q_1, q_2), for all $k \geq 1$

$$P[(\bar{q}_k(1), \bar{q}_k(2)) \geq (q_k(1), q_k(2)) | \ \bar{q}_0(1) = q_0(1) = q_1, \bar{q}_0(2) = q_0(2) = q_2] = 1. \tag{7}$$

Next we show that under (4) and (5), $\{(\bar{q}_k(1), \bar{q}_k(2))\}$ stays bounded with probability 1 for any initial conditions. But considering $\{\bar{q}_k(i)\}$, $i = 1, 2$, we realize it behaves as a GI/GI/1 queue with service times $X_k(i)$ and inter-arrival times $z_k(i)$ whenever $\bar{q}_k(i) > \bar{r}$. Also, with (4) and (5), its traffic intensity $\rho(i) < 1$. Therefore, $\{\bar{q}_k(i)\}$ stays bounded with probability 1. Furthermore, from results on GI/GI/1 queues, the inter-visit times of $q_k(i)$ to the set $\{x : x \leq \bar{r}\}$ have finite mean (this is busy period for the corresponding GI/GI/1 queue).

The above results for $(\bar{q}_k(1), \bar{q}_k(2))$ along with (7) provide the corresponding results for the process $\{(q_k(1), q_k(2))\}$. □

The results of the above theorem do not guarantee existence of a stationary distribution for the process $\{(q_k(1), q_k(2))\}$. However, if we also assume that

$$P[X_k(i) = 0] > 0, \ i = 1, 2 \tag{8}$$

then with some additional work, we can show that the inter-visit time of the process $\{(\bar{q}_k(1), \bar{q}_k(2))\}$ to the set $\{(x, y) : x \leq \bar{r}, y \leq \bar{r}\}$ has a finite mean time. Then from (7) the same is true for the process $\{(q_k(1), \ q_k(2))\}$. Furthermore, the process $\{(q_k(1), q_k(2))\}$ can indeed visit the state (0, 0) with finite mean inter-visit times and the process is an irreducible and aperiodic Markov chain. This guarantees that the Markov chain $\{q_k\}$ has a unique stationary distribution and starting from any initial conditions, it converges to the stationary distribution.

Although (4) and (5) have been shown to be sufficient conditions for stability, for symmetric rate and traffic conditions they are actually also necessary (take $p = \frac{1}{2}$ in this case). To see this, one only has to observe that summing (4) and (5) provides the maximum possible link rates one can obtain for this system. For symmetric case, taking half of the sum provides the largest possible throughput each queue can get. This argument holds for more than two queues (considered below) also.

The conditions (4) and (5) can be immediately generalized to $N \geq 2$ user case: for all i,

$$\mathbb{E}[X(i)] < \mathbb{E}[r(i)1_{\{r(i)>r(j),\forall j\neq i\}}] + \sum_{(j_1,j_2,...j_k)} p(i,j_1,j_2,...,j_k). \\ \mathbb{E}[r(i)1_{\{r(i)=r(j),j=j_1,j_2,...,j_k,r(i)>r(j),j\neq i,j_1,...,j_k\}}] \tag{9}$$

where $p(i, j_1, ..., j_k)$ are the probabilities used for breaking ties and the summation is over all possible subsets of $\{1, ..., N\} - \{i\}$.

Next let us consider the asymmetric case. For simplicity we restrict ourselves to two queues. See the generalization in the next section.

If (say) $\mathbb{E}[X(2)]$ is much smaller than the RHS of (5), then (4) is too conservative for $\mathbb{E}[X(1)]$. At least when $q_k(2) + X_k(2) = 0$, then MAX-TRANS will allow Q_1 to transmit as long as $q_k(1) + X_k(1) > 0$. Thus (4) can be relaxed to

$$\begin{aligned}\mathbb{E}[X_k(1)] < \mathbb{E}[r_k(1)1_{\{r_k(1)>r_k(2)\}}] &+ p\mathbb{E}[r_k(1)1_{\{r_k(1)=r_k(2)\}}] \\ &+ (1-p)\mathbb{E}[r_k(1)1_{\{r_k(1)=r_k(2)\}}1_{\{q_k(2)+X_k(2)=0\}}] \\ &+ \mathbb{E}[r_k(1)1_{\{r_k(1)<r_k(2)\}}1_{\{q_k(2)+X_k(2)=0\}}].\end{aligned} \tag{10}$$

Since $q_k(i) + X_k(i)$ is independent of $r_k(1)$ and $r_k(2)$ the last two expressions in the RHS of the above inequality can be simplified. For example, the last expression becomes $\mathbb{E}[r_k(1)1_{\{r_k(1)<r_k(2)\}}]P(q_k(2) + X_k(2) = 0)$ where $P(q_k(2) + X_k(2) = 0)$ is the probability under stationarity. But we do not know $P(q_k(2) + X_k(2) = 0)$. Motivated by GI/GI/1 queues, we approximate it by $1 - \rho(2)$ where

$$\rho(2) = \frac{\mathbb{E}[X(2)]}{\mathbb{E}[r(2)1_{\{r(2)>r(1)\}}] + (1-p)\mathbb{E}[r(2)1_{\{r(2)=r(1)\}}]}. \tag{11}$$

Finally we provide approximations for mean queue lengths and delays. For simplicity we limit to two queue case. The general case is considered in the next section. First consider $\mathbb{E}[q(i)]$, the mean queue length of the i^{th} queue under steady state. We use the approximation formula for a GI/GI/I queue available in [11]:

$$\mathbb{E}[q] \approx \frac{\rho\ g\ \mathbb{E}[X](C_X^2 + C_r^2)}{2(1-\rho)} \tag{12}$$

where g, C_X^2, C_r^2 are

$$\rho = \mathbb{E}[X]/\mathbb{E}[r], \quad C_X^2 = \frac{\text{var}(X)}{(\mathbb{E}[X])^2}, \quad C_r^2 = \frac{\text{var}(r)}{(\mathbb{E}[r])^2},$$
$$g = \exp\Big[-2\,\frac{1-\rho}{3\rho}\,\frac{(1-C_r^2)^2}{C_r^2+C_X^2}\Big], \quad if \quad C_r^2 < 1,$$
$$= \exp\Big[-(1-\rho)\,\frac{C_r^2-1}{C_r^2+4C_X^2}\Big], \quad if \quad C_r^2 \geq 1.$$

We use statistics for $X(i)$ and $r(i)$ in (12) but the $\rho(i)$ will be as follows. Queue1 at least under heavy traffic receives rate $\mathbb{E}[z_k(1)]$ where

$$\mathbb{E}[z_k(1)] = \mathbb{E}[r_k(1)1\{r_k(1) > r_k(2)\}] + p\mathbb{E}[r_k(1)1\{r_k(1) = r_k(2)\}]. \quad (13)$$

But it will also receive (as argued above for stability) an extra service

$$(1-p)(1-\rho(2))\mathbb{E}[r(1)1_{\{r(1)=r(2)\}}] + \ (1-\rho(2))\mathbb{E}[r(1)1_{\{r(1)<r(2)\}}]. \quad (14)$$

Thus we define $\rho(1)$ for queue1 as $\mathbb{E}[X(1)]$ divided by the sum of (13) and (14). Similarly we define $\mathbb{E}[z_k(2)]$ and $\rho(2)$.

Once $\mathbb{E}[q(i)]$ has been approximated we obtain approximations for $\mathbb{E}[D(i)]$, the mean delay of the first bit arriving at a slot under heavy traffic as (see details in [16])

$$\mathbb{E}[D(i)] = \frac{\mathbb{E}[q(i)]}{\mathbb{E}[z(i)]} + \frac{\mathbb{E}[z(i)^2]}{2(\mathbb{E}[z(i)])^2} + \frac{d}{2\mathbb{E}[z(i)]} \quad (15)$$

where d is the lattice span of the arithmetic distribution of r (and equals 0 if it is non-arithmetic). We will see in the next section that (15) in fact provides a good approximation even under low traffic intensities (we will in fact show it for more than two queues).

Similarly one can get approximation of the mean delay of the last bit arriving at a slot and the mean delay of any bit (average delay of all bits).

5 Simulations

Now we compare the stability results and the approximate mean delays and queue lengths obtained in Section 4 with simulations and verify our claims.

First we consider a system with 10 queues and symmetric statistics. We take $r_k(i) = 25, 15, 0$ with probabilities 0.3, 0.4 and 0.3 respectively. We break the ties assigning a slot with equal probabilities. Then stability boundary from (9) is 2.47. Let $X_k(i)$ take values 5, 2 and 1. The distributions of $X_k(i)$ are the same and we change them so that $\mathbb{E}[X_k(i)]$ is increased from 2.2 to 2.8. For each case simulations were run for 2 million slots. The mean queue lengths of the first five queues are plotted in Figure 1. We see that all the queues start becoming unstable around 2.47 verifying our claim. The same is true for the other five queues also.

Next we consider an asymmetric case with 10 queues. Now the $r_k(i)$ are 50 with probability 0.65 and 0 otherwise. The $X_k(i)$ take values 0 and 10. For $i = 1, 2, 3$, $P[X_k(i) = 10] = 0.2$.

The stability region for the queues $i = 4, ..., 10$ is (approximately) obtained by adding to (9) the extra terms

$$\frac{3(1-\rho)}{2}\mathbb{E}[r(i)1_{\{r(i)=r(1),r(i)>r(j),\text{for all } j\neq 1,i\}}]$$
$$+\frac{3.2.(1-\rho)}{3}\mathbb{E}[r(i)1_{\{r(i)=r(1)=r(2),r(i)>r(j),j\neq,1,2,i\}}]$$
$$+\frac{3}{4}(1-\rho)\mathbb{E}[r(i)1_{\{r(i)=r(1)=r(2)=r(3),r(i)>r(j),j\neq 1,2,3,i\}}].$$

The justification for adding these extra terms is that queues 1, 2, 3 are lightly loaded and hence the other queues can get their share of slots if these are empty. The probability that any of the queues 1, 2, 3 is empty is approximated by $(1-\rho)$ where $\rho = \mathbb{E}[X(1)]/\mathbb{E}[r'(1)]$ and $\mathbb{E}[r'(1)]$ is obtained from (9). Thus the stability region for queues 4, ..., 10 is 6.162.

The $P[X_k(i) = 10]$, $i = 4, ..., 10$ is the same and increased such that $\mathbb{E}[X(i)]$ varies from 5.80 to 6.80.

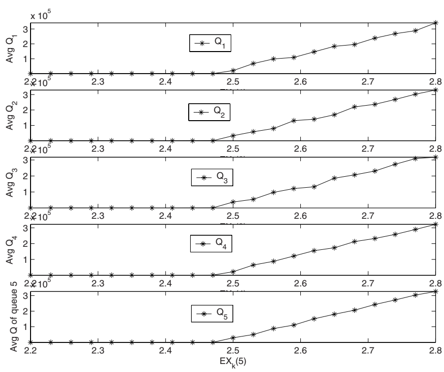

Fig. 1. Average queue length: 10 queue symmetric case

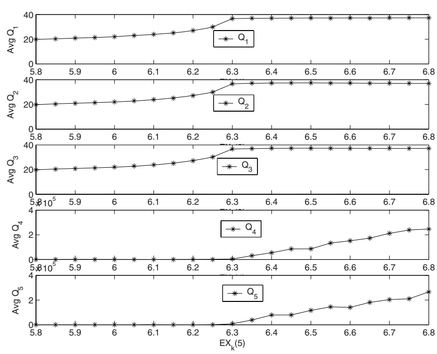

Fig. 2. Average queue length: 10 queue asymmetric case

The plots for $\mathbb{E}[q(i)]$ $i = 1, 2, ..., 5$ are provided in Figure 2. We see that queues 1, 2, 3 stay stable but queues 4 and 5 become unstable at 6.3. The same happens with the queues 6, 7, 8, 9, 10.

Finally we verify the accuracy of the approximate formulae (12) and (15) for mean queue lengths and mean delays. We consider a 2 queue symmetric case. The $r(i)$ take values 10 and 0 with probabilities 0.75 and 0.25. The $X(i)$ take values 15 and 0 and their probabilities are changed to vary the traffic intensity.

The theoretical and simulated values of the mean delays and the mean queue lengths are provided in Table 5. We see that our approximations match the simulated values, quite well.

Table 5. Comparison of theoretical and simulated values of$\mathbb{E}[Q_1]$ and $\mathbb{E}[D_1]$ for 2 queue case

traffic	simulated $\mathbb{E}[Q_1]$	theoretical $\mathbb{E}[Q_1]$	simulated $\mathbb{E}[D_1]$	theoretical $\mathbb{E}[D_1]$ with correction
0.30	1.60	1.68	2.54	2.49
0.50	3.76	3.86	2.90	2.96
0.70	8.82	8.87	3.86	4.03
0.90	33.95	33.87	9.11	9.31
0.95	71.72	70. 78	17.07	17.23
0.98	187.52	182.11	41.76	40.98

Table 6. queue, symmetric case

traffic	simulated $\mathbb{E}[Q_1]$	theoretical $\mathbb{E}[Q_1]$	simulated $\mathbb{E}[D_1]$	theoretical $\mathbb{E}[D_1]$ with correction
0.30	3.86	3.77	3.02	2.94
0.50	8.76	5.88	3.63	3.15
0.70	15.09	11.66	3.99	3.73
0.80	19.76	19.57	4.22	4.53
0.85	23.18	27.84	4.42	5.36

Table 7. queue, asymmetric case

traffic	simulated $\mathbb{E}[Q_1]$	theoretical $\mathbb{E}[Q_1]$	simulated $\mathbb{E}[D_1]$	theoretical $\mathbb{E}[D_1]$ with correction
0.30	6.53	7.38	2.41	1.76
0.50	15.00	16.48	2.73	2.25
0.70	33.87	36.76	3.49	3.33
0.80	59.65	61.58	4.64	4.66
0.85	86.59	86.23	5.89	5.98

Next we verify these formulae for a five queue example. First consider the symmetric case. The $r(i)$ take values 50 and 1 with probabilities 0.65 and 0.35 respectively. The theoretical and simulated $\mathbb{E}[Q_1]$ and $\mathbb{E}[D_1]$ are shown in Table 6 for different $\mathbb{E}[X(i)]$. We see that the approximations are quite reasonable, particularly for $\mathbb{E}[D_1]$.

We also consider a five queue asymmetric case. The $r(i)$ have the same distributions as for the symmetric case. The $X(i)$ take values 40 and 0. We take $P[X(i) = 40] = 0.05$, for $i = 1, 2, 3$, and $P[X(i) = 40]$, $i = 4, 5$ are varied to have different traffic intensities. The values of $\mathbb{E}[Q_4]$ and $\mathbb{E}[D_4]$ are provided in Table 7. We see a good match between theory and simulations.

6 Conclusions

In this paper we considered the problem of optimal scheduling of wireless links. We proved the existence of optimal and average cost optimal policies. Then we identified a greedy policy, called MAX-TRANS which is easy to implement, performs close to optimal and does not require the channel and link rate statistics. Its performance (mean queue lengths and delays) is better than the other well known policies. However, it is not throughput optimal (for asymmetric systems). Thus, we obtained stability region of this policy and approximate mean queue lengths and delays. We also modified this policy to make it throughput optimal.

References

1. Altman, E., Kushner, H.J.: Control of polling in presence of vacations in heavy traffic with applications to satellite and mobile radio systems. SIAM J. Control and Optimization 41, 217–252 (2000)
2. Andrews, M., Kumaran, K., Ramanan, K., Stolyer, A., Vijaykumar, R., Whiting, P.: Scheduling in a queueing system with asynchronously varying service rates. Probability in the Engineering & Informational Sciences 18, 191–217 (2004)
3. Berggren, F., Jantti, R.: Asymptotically fair transmission scheduling over fading channels. IEEE Trans. Wireless Communication 3, 326–336 (2004)
4. Berggren, F., Jantti, R.: Multiuser scheduling over Rayleigh fading channels. Global Telecommunications Conference, IEEE GLOBECOM, PP. 158–162 (2003)
5. Borst, S.: User level performance of channel aware scheduling algorithms in wireless data networks. IEEE/ACM Trans. Networking (TON) 13, 636–647 (2005)
6. Eryilamz, A., Srikant, R., Perkins, J.R.: Stable scheduling policies for fading wireless channels. IEEE/ACM Trans. Networking 13, 411–424 (2005)
7. Goyal, M., Kumar, A., Sharma, V.: A stochastic control approach for scheduling multimedia transmissions over a polled multiaccess fading channel. ACM/SPRINGER Journal on Wireless Networks, to appear (2006)
8. Georgiadis, L., Neely, M.J., Tassiulas, L.: Resource allocation and cross-layer control in Wireless networks. In: Foundations and Trends in Networking, vol. 1, pp. 1–144. New Publishers, Inc, Hanover, MA (2006)
9. Hernandez-Lerma, O., Lassere, J.B.: Discrete time Markov Control Processes. Springer, Heidelberg (1996)
10. Knopp, R., Humblet, P.: Information capacity and power control in single-cell multiuser communication. Proc. IEEE ICC 1995, pp. 331–335 (1995)
11. Krämer, W., Langenbach-Bellz, M.: Approximate formulae for the delay in the queueing system GI/G/1. 8th International Teletrafic Congress, pp. 235–1–8 (1976)
12. Lin, X., Chong, E.K.P., Shroff, N.B.: Opportunistic transmission schdeuling with resource-sharing constraints in Wireless networks. IEEE Journal on Selected Areas in Communications 19, 2053–2064 (2001)
13. Lin, X., Shroff, N.B., Srikant, R.: A tutorial on cross-layer optimization in wireless networks. IEEE Journal on Selected Areas in Communications 24, 1452–1463 (2006)
14. Lin, P., Berry, R., Honig, M.: A Fluid Analysis of a Utility-based Wireless Scheduling Policy, to appear in IEEE Transaction on Information Theory
15. Muller, A., Stoyan, D.: Comparison methods for stochastic Models and Risks. Wiley, Chichester, England (2002)
16. Prasad, D.K.: Scheduling and performance analysis of queues in wireless systems M.E. Thesis, Dept. of Electrical Communication Engg., Indian Institute of Science, Bangalore (July 2006)

17. Shakkottai, S., Stolyar, A.: Scheduling algorithms for a mixture of real-time and non-real time data in HDR. "citeseer.ist.edu/466123.html"
18. Shetiya, H., Sharma, V.: Algorithms for routing and centralized scheduling in IEEE 802.16 mesh networks. In: Proc IEEE WCNC (2006)
19. Tassiulas, L., Ephremides, A.: Dynamic server allocation to parallel queues with randomly varying connectivity. IEEE Trans., Information Theory 39, 466–478 (1993)
20. Tsibonis, V., Georgiadis, L., Tassiulas, L.: Exploiting wireless channel state information for throughput maximization. IEEE Trans. Inf. Theory 50, 2566–2582 (2004)
21. Walrand, J.: An Introduction to Queueing Networks. Printice Hall, Englewood Cliffs (1988)