Opportunistic Splitting Algorithms For Wireless Networks With Heterogeneous Users

Xiangping Qin and Randall Berry
Dept. of Electrical and Computer Engineering
Northwestern University
2145 Sheridan Rd., Evanston IL 60208
Email: {sandra,rberry}@ece.northwestern.edu

***Abstract*— In this paper, we develop medium access control protocols to enable users in a wireless network to opportunistically transmit when they have favorable channel conditions, without requiring a centralized scheduler. We consider approaches that use splitting algorithms to resolve collisions over a sequence of mini-slots to determine which user will transmit. We consider an asymmetric model where different users may have different channel statistics and different priority levels. In this context, we propose a new fairness criterion, *distribution fairness*, that lends itself naturally to a distributed implementation. We give a splitting algorithm for achieving distribution fairness. We analyze the performance of this algorithm and show that the average overhead required is less than in a symmetric case, which we have analyzed previously. Simulation results are also given.**

I. Introduction

Recently, "opportunistic scheduling" approaches have received much attention as a means for exploiting the "multiuser diversity" inherent in a wireless setting (e.g., [1–3]). These approaches attempt to schedule transmissions during periods when a user's channel is "good" and hence can support a larger transmission rate. This has a theoretical basis in work such as [7], which shows that to maximize the ergodic capacity of a multiple-access fading channel, at most a single user with the best channel state should transmit at any time. For an up-link (multiple access) model, such approaches require a centralized scheduler with knowledge of each user's channel gain to select the user to transmit at a given time-slot. This requires the scheduler to acquire estimates of each users' channel state before making the scheduling decision; the overhead and delay incurred in doing this may limit the system's performance, particularly if the number of active users is large or the channels change rapidly.

In [5] [4], we have consider distributed approaches for opportunistic scheduling where each user has knowledge of its own channel conditions, but no knowledge of the other users' channels. The transmission decisions are individually made by each user based on their local channel information. This approach requires less overhead and scales well as the number of users increases. In [5], a *channel-aware Aloha* approach is introduced, where users base their transmission probabilities on their channel gain. Similar approaches have also been studied in [6]. In [5], it is shown that the total throughput increases with the number of users at the same rate as in the optimal centralized scheme, but is asymptotically reduced by a factor of $1/e$ due to the contention. In [4], a distributed approach is given based on using splitting algorithms [10] to determine the user with the best channel over a sequence of mini-slots. For a homogeneous model where the users have identical channel statistics, it is shown that the average number of mini-slots required to find the user with the best channel is less than 2.5 independent of the number of users or the fading distribution. In other words, the overhead needed for this type of approach scales well as the number of users increases.

In both [5] and [4], we considered a homogeneous model, where each user's channel gains were independent and identically distributed. In this case, a scheduling rule that maximizes the total throughput results in each user having an equal throughput. In practice, the set of users will likely have asymmetric channel statistics, for example due to differences in location or mobility. In this paper, we consider distributed approaches for opportunistic transmission in this type of heterogeneous model.

In a heterogeneous setting, an important issue to how to guarantee some level of fairness among the users. In particular, simply maximizing the total rate as in [4], [5], will tend to overly favor users with better channel statistics. Also in certain cases, it may be desirable to give some of the users a larger share of the systems resources than others. Here we address these concerns by using a new type of fairness called *distribution fairness*, that naturally leads to a distributed implementation as in [4]. With this definition, each user is guaranteed to be able to transmit for a specified fraction of time. Given this fraction of time, the user is allowed to transmit during those times, when its channel is statistically "better" than average. We give a precise definition of this in terms of an optimization formulation, that is related to work in [1]. In [1], the goal is to maximize a total system utility under a constraint on the fraction of time each user transmits. In the formulation for distribution fairness, a users "utility" is defined in terms of its channel distribution. The aim of this type of fairness is similar to that of the proportional fair scheduling rule in [3], where the user who has a higher ratio of transmission rate to its average transmission rate is scheduled to transmit.

This research was supported in part by the Motorola-Northwestern Center for Telecommunications and NSF CAREER award CCR-0238382.

With proportional fairness the ratio of the average throughput of any two users is fixed, while the fraction of time each user can transmit may vary. Another related fairness criterion was given in [9]. With the distribution fairness, the ratio of the fraction of time different user transmits can be adapted to emulate other types of fairness criterion as well.

In the following section, we give a precise definition of distribution fairness and discuss its properties in more detail. When then given similar splitting algorithms to those in [4] for achieving distribution fairness in a distributed setting. We show that in certain cases the number of mini-slots required to resolve collisions in this asymmetric model is no greater than those required in a symmetric model as in [4]. Analysis and simulation results are both presented.

II. Distribution fair scheduling

We consider a model of the up-link in a wireless network with N users all transmitting to a common receiver. The channel between each user and the receiver is modeled as a time-slotted, block-fading channel; if only the ith user transmits in a given time-slot, the received signal, $y_i(t)$ is given by

$$y_i(t) = \sqrt{H_i} x_i(t) + z(t),$$

where $x_i(t)$ is the transmitted signal, H_i is the fading channel gain, and $z(t)$ is additive white Gaussian noise. The channel gain is assumed to be fixed during each time slot and to randomly vary between time-slots. In the following, we assume that the channel gains of each user in each time-slot are independent random variables, with probability density functions $f_{H_i}(h_i)$ on $[0, \infty)$ for $i = 1, \dots, N$. Let $\bar{F}_{H_i}(h_i) = \int_h^\infty f_{H_i}(h_i)\, dh$ denote the complimentary distribution function for user i's channel gain. To be begin, we consider a centralized TDM scheduler which, given the vector of channel gains $\mathbf{h} = (h_1, h_2, ..., h_N)$ at each time-slot, schedules one of the users to transmit. Let $A(\mathbf{h})$ denote scheduling allocation, i.e. $A(\mathbf{h}) = i$ if user i is scheduled when the joint channel state is $\mathbf{h}$. We assume that all users are infinitely back-logged, and focus on the average throughput achieved by each user.

Definition: An allocation $A(\mathbf{h})$ $\mathbf{h} = (h_1, h_2, ..., h_N)$ is defined to be *distribution fair* with parameters $p_1, p_2, ..., p_N$, if it satisfies

$$\min_{A(\mathbf{h})} E_H \left(\sum_{i=1}^{N} \bar{F}_{H_i}(h_i) 1_{A(\mathbf{h})=i} \right) \tag{1}$$
$$\text{subject to:} \quad \text{Prob } \{A(\mathbf{h}) = i\} = p_i \;\; \text{for } i = 1, \dots, N.$$

Here,

$$1_{A(\mathbf{h})=i} = \begin{cases} 1, & \text{if A(}\mathbf{h}\text{)=i,} \\ 0, & \text{otherwise.} \end{cases}$$

The parameters $p_1, \dots, p_N$ should be a probability mass function and specify the fraction of time that each user will transmit. Given this constraint, a distribution fair scheduler will attempt to schedule users with a small value of $\bar{F}_{H_i}(h_i)$; this corresponds to a user having a channel that is statistically strong, relative to the users own distribution. This optimization problem is similar to that in [1] and the solution is a simple extension of [1]. The difference is that in [1], the goal is to optimize the sum expected utility, while the problem here it is to minimize the sum of the values of the complimentary distribution function, which can be viewed as a special utility function. Using the results in [1], the optimal solution to (1) is given by

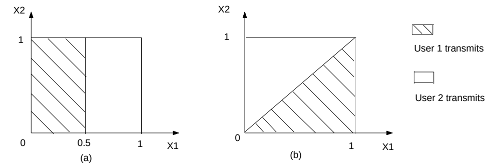

Fig. 1. Comparison of different scheduling schemes as a function of $X_1 = \bar{F}_{H_1}(h_1)$ and $X_2 = \bar{F}_{H_2}(h_2)$.

$$A(\mathbf{h}) = \arg \min_{i=1..N} (\bar{F}_{H_i}(h_i) + v_i), \tag{2}$$

where the parameters v_i are chosen to satisfy the constraints Prob$\{A(\mathbf{h}) = i\} = p_i$, and so can be interpreted as "fairness parameters." For convenience, we set $v_i = 0$ for $i = \max_{i=1..N} p_i$, then it can be shown that $v_j \geq 0$, for all j.

Next we compare the performance of such a scheduling rule with several other alternatives. First suppose that instead of (1) we maximize the expected total throughput, i.e.

$$E_H \left(\sum_{i=1}^{N} R(h_i) 1_{A(\mathbf{h})=i} \right),$$

given the same constraints, where $R(\cdot)$ gives the transmission rate a a function of the channel gain. This type of scheduling policy was considered in [1] and will tend to favor the user with the better channel distribution more than the distribution fair approach. For example, assume that there are 2 users, such that $R(h_1) > R(h_2)$ for any h_1 that satisfies $\bar{F}_{H_1}(h_1) < 0.5$ and for any h_2. If the goal is to maximize the total throughput with the constraint that $p_1 = p_2 = 1/2$, then the optimal scheduling scheme will be as shown in Fig. $1(a)$. This figure shows the optimal scheduling decision as a function of $X_i = \bar{F}_{H_i}(h_i)$, for each user i. In this case, user 1 only transmits when its gain satisfies $\bar{F}_{H_{th}}(h_1) \leq 0.5$, i.e. when it has a strong channel. However, the weaker user is likely to transmit in any channel state. For this case, the distribution fair scheduling policy is shown in Fig. $1(b)$ where now both users are more likely to transmit when their channel is strong relative to its own statistics.

Next we compare a distribution fair scheduler to a proportional fair one. We simulated both scheduling rule for two users with independent Rayleigh fading. User 1's channel gain distribution is $f_{H_1}(h_1) = \frac{1}{h_{01}} \exp(\frac{-h_1}{h_{01}})$ and user 2's is $f_{H_2}(h_2) = \frac{1}{h_{02}} \exp(\frac{-h_2}{h_{02}})$, where the average channel gains are $h_{01} = 1$ and $h_{02} = 0.5$. Fig. 2 shows the average transmission rate for both users as well as the sum rate. It can be seen that both scheduling rules have similar performance. With the

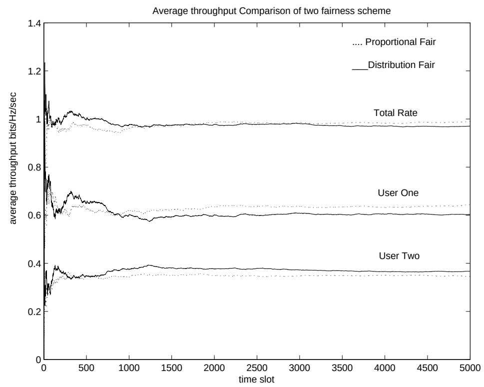

Fig. 2. Average transmission rate of each users.

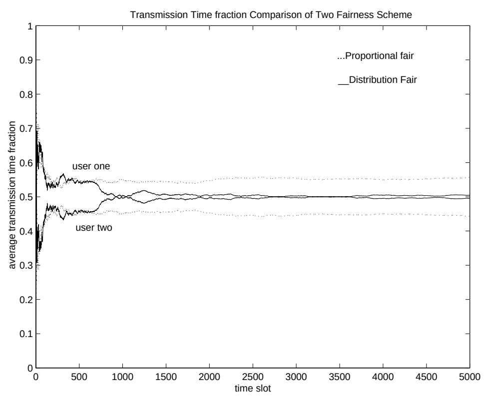

Fig. 3. Fraction of time transmitting to each user.

distribution fair scheduler the average rate for user 1 (2) is slightly lower (higher) compared to the proportional fair case; the total throughput is slightly lower than in the proportional fair case. Fig. 3 shows the average fraction of time each user transmits. The distribution fair scheduler guarantees an equal portion of time to each user. In this case, the proportional fair scheduler alots user 1 a larger portion of time than user 2, which also helps to explain why the proportional fair scheduler achieves a higher throughput.

In the remainder of the paper, we only consider two classes of users. Where all the users in a given class have i.i.d. channel gains and are assigned the same p_i in (1) (and thus the same fairness parameter v_i). For $k = 1, 2$, let $\mathcal{C}_k$ denote the set of class k users, and let $n_k = |\mathcal{C}|_k$ be the number of class k users. Without loss of generality, we assume that $v_i = 0$ for all users $i \in \mathcal{C}_1$, and that $v_i = v > 0$ for all $i \in \mathcal{C}_2$. In this case, the scheduling rule in (2) can be viewed as finding the minimum of of a set of random variables $\{X_i\}$, where for each $i \in \mathcal{C}_1$, $X_i = \bar{F}_{H_i}(H_i)$ is an i.i.d. uniform random variable on $[0, 1]$, and for each $i \in \mathcal{C}_2$, $X_i = \bar{F}_{H_i}(H_i) + v$ is an i.i.d. uniform random variable on $[v, 1 + v]$. Also, let $P_k = \Pr(A(\mathbf{H}) \in \mathbb{C}_k)$ for $k = 1, 2$, where $\mathbf{H} = \{H_i\}$ is random vector of channel gains chosen according to the joint density, $\prod_i f_{H_i}(h_i)$. Thus, P_k is the probability that a class k user is scheduled. Assuming the scheduler is non-idling, we have $P_1 + P_2 = 1$. From the above, P_2 is given by

$$P_2 = (1 - v_1)^{n_1} \sum_{i=1}^{n_2} \binom{n_2}{i} (1 - v_1)^i v_1^{n_2 - i} \left(\frac{i}{i + n_1} \right).$$

Next we give tight upper and lower bounds on P_2; this can, in turn, be used to help calculate the correct fairness parameter v for a given choice of p_1 and p_2.

Proposition 1: The probability a class 2 user is scheduled, P_2 satisfies:

$$\frac{(1 - v_1)^{n_1+1} n_2}{(1 - v_1) n_2 + n_1} \geq P_2 \geq \frac{(1 - v_1)^{n_1+1} n_2}{(1 - v_1) n_2 + n_1 + v_1}.$$

Proof: First we derive the upper bound. Note that

$$\begin{aligned} &\sum_{i=1}^{n_2} \binom{n_2}{i} (1 - v_1)^i v_1^{n_2 - i} \frac{i}{i + n_1} \\ &= \sum_{i=0}^{n_2} \binom{n_2}{i} (1 - v_1)^i v_1^{n_2 - i} \frac{i}{i + n_1} \\ &= \mathbb{E}\left[\frac{I}{I + n_1} \right], \end{aligned}$$

where the expectation is with respect to the p.m.f. $q_i = (1 - v_1)^i v_1^{n_2 - i}$. From Jenson's inequality this is upper bounded by:

$$\frac{\mathbb{E}(I)}{\mathbb{E}(I) + n_1} = \frac{(1 - v_1) n_2}{(1 - v_1) n_2 + n_1}.$$

Therefore,

$$P_2 \leq \frac{(1 - v_1)^{n_1+1} n_2}{(1 - v_1) n_2 + n_1}.$$

For the lower bound, note that

$$\binom{n_2}{i} i = n_2 \binom{n_2 - 1}{i - 1}.$$

Thus, we have

$$\begin{aligned} &\sum_{i=1}^{n_2} \binom{n_2}{i} (1 - v_1)^i v_1^{n_2 - i} \frac{i}{i + n_1} \\ =& n_2 \sum_{i=1}^{n_2} \binom{n_2 - 1}{i - 1} (1 - v_1)^i v_1^{n_2 - i} \frac{i}{i + n_1} \\ =& n_2 \sum_{i'=0}^{n_2'} \binom{n_2'}{i'} (1 - v_1)^{i'} v_1^{n_2' - i'} \frac{(1 - v_1)}{i' + n_1 + 1}, \end{aligned}$$

where $n_2' = n_2 - 1$. This is lower bounded by

$$\begin{aligned} &n_2 \frac{1 - v_1}{(1 - v_1)(n_2 - 1) + n_1 + 1} \\ =& \frac{(1 - v_1)^{n_1+1} n_2}{(1 - v_1) n_2 + n_1 + v_1} \end{aligned}$$

The lower bound then follows. ■

When $v_1 << \frac{n_1 + n_1}{1 + n_2}$, the two bounds are very close to each other; this will be true for most values of n_1 and n_2 as shown in Fig. 4. For each user of class 2, the probability of transmission is $\frac{P_2}{n_2}$. The probability that the users of class 1 transmit is $1 - P_2$, and the transmission probability for each user of class 1 is $\frac{1 - P_2}{n_1}$. Therefore for a given P_1 and P_2, v can be found accordingly.

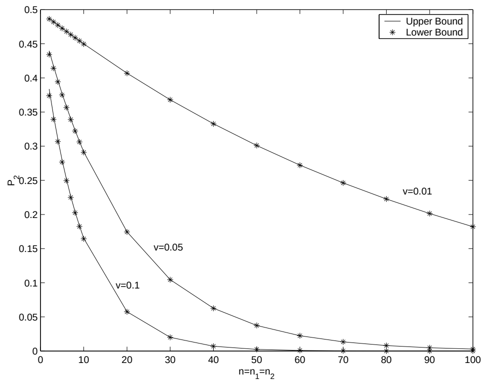

Fig. 4. Upper and lower bounds on P_2

III. A Distributed Splitting Algorithm

In this section, we consider a distributed approach for implementing a distribution fair scheduling policy. Our approach is a generalization of the splitting algorithm given in [4]. We consider a slotted system where the channel gain is a constant within one slot. In each slot, each user knows only their own channel gain during the slot, but not the gain of any other user. At the beginning of each slot, mini-slots are used for users to send requests to the receiver. Users will send a request if their channel exceeds a threshold. The receiver will feedback $(0, 1, e)$ indicating the mini-slot is idle, successful or a collision happened. Based on this feedback, the user will adjust their threshold and either send a request again or back-off, until a successful request is received. The successful user will then continue to transmit data for the rest of the slot. We are still consider a 2 class system as above, and we assume that the offset for class 2 v in (2is known by all class 2 users. We then consider using a splitting algorithm to find the user that satisfies (2). First, we assume that after each mini-slot, the number of users of each class involved in a collision, n_1 for class one and n_2 for class two, is known. We will remove this assumption later.

The splitting algorithm in this case is given by the following:

initialize: $m = 0$, $H_h = \infty$, $H_l = h_l^{init}$ and $H_{ll} = 0$
while $m \neq 1$ and $k \leq K$ **do**
  $m =$ (0,1,e) (feedback from the base station).
  if $m = e$ **then**
    Base station feeds back (m_1, m_2), where m_i is number of class i users involved in collision.
  end if
  if $m = e$ **then**
    $H_{ll} = H_l$; $H_l = \text{split}(H_l, H_h, m_1, m_2)$;
  else if $m = 0$ **then**
    if $H_{ll} \neq 0$ **then**
      $H_l' = \text{split}(H_{ll}, H_l, m_1, m_2)$;
    else
      $H_l' = \text{split}(H_{ll}, H_l, n_1, n_2)$;
    end if
    $H_h = H_l$;
    $H_l = H_l'$;
  end if
  $k = k + 1$
end while

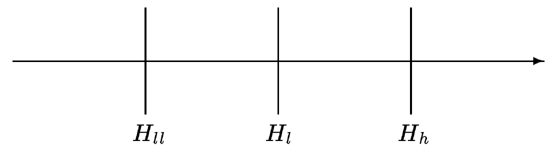

Fig. 5. Example of a split range: H_{ll} is largest value of H_l used in the prior mini-slots such that there are some users above H_{ll}. $H_l < H < H_h$ is the transmission range.

At each stage, only the users whose channel gains are between H_l and H_h will transmit. H_{ll} denotes the maximum channel gain above which it is known that there is at least one user. As shown in Fig. 5, if a collision occurs ($m = e$), the range $H_l < h < H_h$ is split into two parts (denoted by the function "split"); users in the upper part will transmit in the next mini-slot. If an idle mini-slot occurs ($m = 0$), there are two possibilities: One, as shown in Figure 5 is that there has been a collision before, *i.e.* $H_{ll} \neq 0$. This means that the best channel gain lies between $H_{ll} < h < H_l$. The other possibility is that there has never been a collision before, *i.e.* $H_{ll} = 0$. This means all the users' channel gains are all below H_l. In both cases we split the interval $[H_{ll}, H_l]$ into two parts; the new transmission range will be the upper part.

The function $\text{split}(H_l, H_h, n_1, n_2)$ is chosen to minimize the number of mini-slots required. In [4] it is shown that for $v = 0$, the average number of mini-slots required is less than 2.4414 when the number of users involved in a collision is known. In the following we consider whether the average number of mini-slots required for this asymmetric model with $v \neq 0$ can be no larger than for the $v = 0$ case.

For a given v, let $E(l, v)$ denote the event that for exactly one user i, $X_i \leq l$, where for each i, X_i is a uniform random variable as defined above. This corresponds to a success occurring in the first mini-slot, if $h_l^{init} = l$. The next proposition states that for $v > 0$ the probability of this event for the "best" choice of l will be no less than in the $v = 0$ case.

Proposition 2: Let $P(l, v) = \Pr\{E(l, v)\}$. This satisfies:

$$\max_l P(l, v) \geq \max_l P(l, 0),$$

where $\max_l P(l, 0) = (1 - \frac{1}{n_1+n_2})^{n_1+n_2-1}$.

Proof: By definition, for $l < 1$,

$$P(l, v) = \begin{cases} n_1 l (1-l)^{n_1-1}(1-l+v)^{n_2} \\ \quad +n_2(l-v)(1-l+v)^{n_2-1}(1-l)^{n_1} \\ \qquad \text{for } l > v \\ n_1 l (1-l)^{n_1-1} \ \text{for } l \leq v \end{cases} \qquad (3)$$

First assume $v \geq \frac{1}{n_1}$ and choose

$$l = \frac{1}{n_1}, \tag{4}$$

then $P(l,v) = (1-\frac{1}{n_1})^{n_1-1}$, and since $(1-\frac{1}{x})^{x-1}$ is decreasing,

$$P(l,v) > (1-\frac{1}{n_1+n_2})^{n_1+n_2-1}.$$

Therefore

$$\max P(l,v) \geq (1-\frac{1}{n_1+n_2})^{n_1+n_2-1},$$

for any $v \geq \frac{1}{n_1}$. Next, consider $0 < v < \frac{1}{n_1}$. Choose

$$l = \frac{1+vn_2}{n_1+n_2}, \tag{5}$$

then $l > v$, and so

$$\begin{aligned} P(l,v) =& (1-\frac{1+vn_2}{n_1+n_2})^{n_1-1}(1-\frac{1+vn_1}{n_1+n_2})^{n_2-1} \\ & [(1+vn_2)(1-\frac{1-vn_1}{n_1+n_2})-n_2 v]. \end{aligned} \tag{6}$$

Thus to show $\max_l P(l,v) \geq (1-\frac{1}{n_1+n_2})^{n_1+n_2-1}$, it is sufficient to show that

$$\begin{aligned} &(1-\frac{vn_2}{n_1+n_2-1})^{n_1-1}(1+\frac{vn_1}{n_1+n_2-1})^{n_2-1} \\ &\cdot(1+\frac{v(n_1-n_2)+v^2n_1n_2}{n_1+n_2-1}) \geq 1 \end{aligned} \tag{7}$$

Let

$$\begin{aligned} f(v) =& (n_1-1)\log(1-\frac{vn_2}{n_1+n_2-1}) \\ &+(n_2-1)\log(1+\frac{vn_1}{n_1+n_2-1}) \\ &+\log(1+\frac{v(n_1-n_2)+v^2n_1n_2}{n_1+n_2-1}), \end{aligned}$$

then (7) is equivalent showing that $f(v) > 0$. Taking the derivative of $f(v)$, it can be shown that for $0 < v < \frac{1}{n_1}$, $\frac{\partial f(v)}{\partial v} > 0$. Because $f(0) = 0$, therefore $f(v) > 0$ for $0 < v < \frac{1}{n_1}$, as desired. ■

Let $m(v,K)$ denote the average number of mini-slots required as a function of the offset v and the maximum number of mini-slots, K, *i.e.* if the collision is not resolved after K slots, we stop and restart. We want to show that $m(v,K) \leq m(0,K)$. The following is a proof for the case when there are only two users, one in $\mathcal{C}_1$ and one in $\mathcal{C}_2$. We conjecture this holds in general, but do not yet have a proof.

Corollary 1: If $n_1 = n_2 = 1$, then $m(v,K) \leq m(0,K)$.

Proof: Define a sequence of "stages" $m = 1..K$, where each stage corresponds to one splitting round. At each stage m, we define the system to be in one of three possible "states": $S_m = 0, a, b$. State $S_m = 0$ means a success has occurred in some previous stage and splitting algorithm has stopped. State $S_m = a$ means that at the end of the previous split it is known that both users channel gains are less than some finite value, while state $S_m = b$ means that this is not the case. And if $S_m = a$ or b, the splitting algorithm continues. For each stage, we also define a cost-to-go $J(S_m)$ which is the expected number of additional splitting rounds needed to find the best user starting in state S_m. We assume that after K steps the sequence stops. Therefore for the last stage $J(S_K) = 0$, and for $m \leq K$,

$$J(S_{m-1}) = \begin{cases} 0, & \text{if } S_{m-1} = 0, \\ 1 + \sum_{S_m} P(S_{m-1}, S_m) J(S_m), & \text{o.w.} \end{cases} \tag{8}$$

Here $P(S_{m-1}, S_m)$ is the probability of transitioning from state S_m to S_{m-1}. The Corollary can then be proved using backward induction. Specifically, at stage K, $J(S_K)$ will be the same for both cases. Let $J_n(S_m)$ and $P(S_{m-1}, S_m)$, be the cost-to-go and transition probability for the $v > 0$ case and let $J_n'(S_m)$, $P'(S_{m-1}, S_m)$ be the analogous quantities for the $v = 0$ case. It can be shown that $P(S_{m-1}, S_m) < P'(S_{m-1}, S_m)$, for all $S_{m-1}, S_m = a, b$. From this it follows that $J(S_{m-1}) < J'(S_{m-1})$. By induction, we have $J(S_0) < J'(S_0)$, which means the average number of mini-slots required is less in the asymmetric model than the symmetric case. ■

Based on (4) and (5), we define the function $\text{split}(H_l, H_h, n_1, n_2)$ to be equal to the value of H_l' that satisfies, $\frac{F_H(H_l')-F_H(H_h)}{F_H(H_l)-F_H(H_h)} = g(n_1,n_2,v)$, where

$$\begin{aligned} & g(n_1,n_2,v) \\ =& \begin{cases} \frac{1+(v-F_H(H_h))n_2}{n_1+n_2}, & \text{if } F_H(H_h) < v \ \& \ \frac{n_1(v-F_H(H_h))}{F_H(H_l)-F_H(H_h)} < 1 \\ \frac{1}{n_1+n_2}, & \text{if } F_H(H_h) > v \\ \frac{1}{n_1}, & \text{otherwise.} \end{cases} \end{aligned} \tag{9}$$

Therefore,

$$\begin{aligned} \text{split}_k(H_l, H_h, n_1, n_2) =& F_H^{-1}\left(F_H(H_l) g(n_1,n_2,v)\right. \\ & \left. + (1-g(n_1,n_2,v)) F_H(H_h)\right). \end{aligned}$$

Numerical results using this algorithm are shown in Fig. 6.

Until now we have assumed that after a collision occurs, the number of users of each class involved in the collision is known to all the users. However, this may be hard to realize in practice. Therefore, we next consider the case where the numbers of users involved in a collision is unknown. When a collision occurs, the most likely scenario is that two users were involved in this collision [10]. Accordingly, the splitting algorithm is modified to the following:

initialize: $m = 0$, $H_h = \infty$, $H_l = 0$ and $H_{ll} = 0$
while $m \neq 1$ and $k \leq K$ **do**
  m = (0,1,e) feedback from the base station.
  if $m = e$ **then**
    $H_{ll} = H_l$; $H_l = \text{split}_2(H_l, H_h)$;
  else if $m = 0$ **then**
    if $H_{ll} \neq 0$ **then**
      $H_l' = \text{split}_2(H_{ll}, H_h)$;
    else

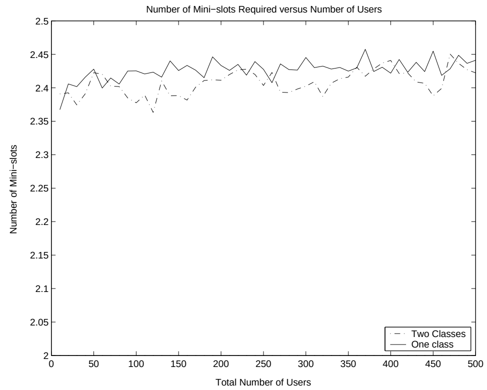

Fig. 6. Number of mini-slots required vs. total number of users with knowledge of number of users of each class involved in a collision

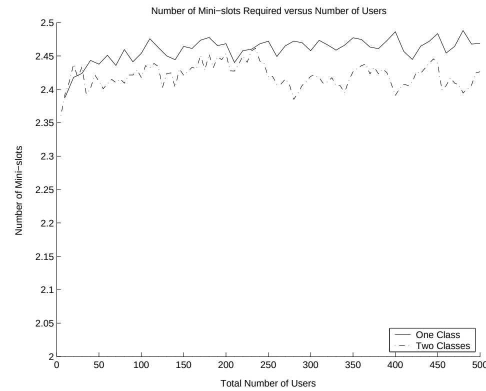

Fig. 7. Number of mini-slots required vs. total number of users without knowledge of the number users of each class involved in a collision

$H_l' = \text{split}(H_l, H_h, k_1, k_2)$;
end if
$H_h = H_l$;
$H_l = H_l'$;
end if
$k = k + 1$
end while

Where

$$\text{split}_2(H_l, H_h) = F_H^{-1}\left(\frac{F_H(H_l) + F_H(H_h)}{2}\right).$$

Again, we want to compare the average number of mini-slots required using this algorithm for two classes to that required for a one class scheme. In [4] we have shown that for the one class scheme this number is no greater than 2.5070. We make the following conjecture. As before, when $n_1 = n_2 = 1$, it can be shown to be true.

Conjecture 1: Without knowing the number of users involved in a collision, the mini-slots required using the given algorithm for two classes is less than the mini-slots required for one class scheme 2.5070.

Next we give some numerical results for both of these algorithms. For the two class case, the same number of users are in each class and $v = 0.01$. Here we generated the normalized random variables $X_i = F_{H_i}(h_i)$ which could correspond to any fading distribution. Fig. 6 shows the average number of mini-slots required when each user learns how many users of each class were involved in a collision; both a single class and a two class case is shown. It can be seen that the mini-slots required for two classes is less than for one class of users as conjectured. The analogous results are shown in Fig. 7 when the number of users involved in the collision is not known by each user. Again, the two class case requires fewer mini-slots.

IV. Conclusion and Future work

In this paper, we introduced a fairness criteria, distribution fairness, and developed distributed splitting algorithms for achieving this. These splitting algorithms were analyzed for a two class case and simulation results are given. In our future work, we would like to complete the analysis of the single class system and extend the model to more than two classes.

References

[1] X. Liu, E. Chong, and N. Shroff, “Opportunistic Transmission Scheduling with Resource-Sharing Constraints in Wireless Networks,” IEEE Journal on Selected Areas in Communications, vol. 19, no. 10, pp. 2053-2064, October, 2001.

[2] S. Borst, “ User Level Performance of Channel-Aware Scheduling Algorithms in Wireless Data Networks”, *Proc. IEEE INFOCOM’03* April, 2003

[3] P. Viswanath, D. Tse and R. Laroia, “opportunistic beamforming using dumb antennas,” IEEE Tran. on Information Theory, vol. 48, no.6, pp. 1277-1294, June 2002.

[4] X. Qin and R. Berry, “Opportunistic Splitting Algorithms For Wireless Networks,” IEEE INFOCOM 2004, Hongkong, March 7-11, 2004.

[5] X. Qin and R. Berry, ”Exploiting Multiuser Diversity for Medium Access Control in Wireless Networks,” IEEE INFOCOM 2003, San Francisco, March 2003.

[6] P. Venkitasubramaniam, S. Adireddy, and L. Tong, “Opportunistic Aloha and Cross-layer Design for Sensor Networks,” *Proc. of IEEE MILCOM*, Oct. 2003, Boston, MA.

[7] R. Knopp and P. A. Humblet, ”Information capacity and power control in single-cell multiuser communications,” *Proc. IEEE ICC ’95*, Seattle, WA, June 1995.

[8] P. Bender., *et al.*, ”CDMA/HDR: a bandwidth efficient high speed wireless data service for nomadic users,” *IEEE Communications Magazine*, Vol. 38, No. 7, pp. 70-77, July 2000.

[9] Jun Sun, Lizhong Zheng and Eytan Modiano, ”Wireless Channel Allocation Using an Auction Algorithm,” Allerton Conference on Communications, Control and Computing, October, 2003.

[10] D. Bertsekas and R. Gallager, ‘*Data Networks*, 2nd Ed., Prentice Hall, 1992.