Multicast Scheduling in Cellular Data Networks

Hyungsuk Won, Han Cai, Do Young Eun, Katherine Guo, Arun Netravali, Injong Rhee, and Krishan Sabnani

Abstract

Multicast is an efficient means of transmitting the same content to multiple receivers while minimizing network resource usage. Applications that can benefit from multicast such as multimedia streaming and download, are now being deployed over 3G wireless data networks. Existing multicast schemes transmit data at a fixed rate that can accommodate the farthest located users in a cell. However, users belonging to the same multicast group can have widely different channel conditions. Thus existing schemes are too conservative by limiting the throughput of users close to the base station. We propose two proportional fair multicast scheduling algorithms that can adapt to dynamic channel states in cellular data networks that use time division multiplexing: Inter-group Proportional Fairness (IPF) and Multicast Proportional Fairness (MPF). These scheduling algorithms take into account (1) reported data rate requests from users which dynamically change to match their link states to the base station, and (2) the average received throughput of each user inside its cell. This information is used by the base station to select an appropriate data rate for each group. We prove that IPF and MPF achieve proportional fairness among groups and among all users in a group inside a cell respectively. Through extensive packet-level simulations, we demonstrate that these algorithms achieve good balance between throughput and fairness among users and groups.

Index Terms

Multicast Scheduling, 3G, Downlink Schedule, Optimization, Proportional Fair, Cellular Data Networks

This is an extended version of the same titled paper appeared in Proceedings of IEEE INFOCOM 2007.

Hyungsuk Won is with Dept. of Computer Science, NC State University. Email: hwon@ncsu.edu.
Han Cai is with Dept. of Electrical and Computer Engineering, NC State University. Email: hcai2@ncsu.edu.
Do Young Eun is with Dept. of Electrical and Computer Engineering, NC State University. Email: dyeun@ncsu.edu.
Katherine Guo is with Bell Laboratories, Alcatel-Lucent. Email: kguo@bell-labs.com.
Arun Netravali is with Bell Laboratories, Alcatel-Lucent. Email: ann@bell-labs.com.
Injong Rhee is with Dept. of Computer Science, NC State University. Email: rhee@ncsu.edu.
Krishan Sabnani is with Bell Laboratories, Alcatel-Lucent. Email: kks@bell-labs.com.

I. Introduction

As of July 2006, the number of CDMA2000 1xEV-DO subscribers has exceeded 38.5 million [1]. Third-generation (3G) wireless data networks support high data rates, e.g. 2.4 Mbps for CDMA2000 1x Evolution-Data Optimized (1xEV-DO) [2] and up to 14.4 Mbps for UMTS High-Speed Downlink Packet Access (HSDPA) [3], enabling a broader range of bandwidth-intensive services. These include streaming media such as the MobiTV service [4] from Sprint, Cingular and Alltel, and the VCast service [5] from Verizon. More sophisticated services, ones which incorporate location information, e.g., live regional traffic reports, geographically targeted advertisements, are expected next. A key factor distinguishing these new applications is that they are naturally amenable to multicast transmission from the base station in a cell.

There has been much research on unicast scheduling in cellular networks (e.g., [2], [6], [7], [8], [9], [10], [11], [12], [13]). Typically, these systems employ Time Division Multiplexing (TDM) in the downlink direction; real time is divided into small fixed time slots. For example, CDMA2000 1xEV-DO downlinks use TDM with a time slot length of 1.67 ms. During different time slots, each user may experience a different signal-to-noise ratio (SNR), which determines the maximum rate at which this user can receive data reliably. For unicast, for each slot, each user sends the Data Rate Control (DRC) message specifying this maximum rate back to the base station. Since each user can specify a different DRC, the *unicast* scheduler at the base station needs to decide on *which user* to serve at each slot based on user DRC feedback. Once the base station selects a user, it transmits to the user using a modulation and error coding scheme suitable for its DRC rate. Note that if the base station sends at a rate higher than the user DRC rate, then the user *cannot* receive *any* data. State of the art unicast schedulers exploit channel states of users to increase overall throughput; usually by favoring transmissions to users with high DRC rates.

What makes the design of a multicast scheduler different from unicast? In multicast, at each time slot, the base station can transmit only to *one* multicast group at *one* rate. There are multiple multicast groups in a cell and each multicast group may contain a different number of users which might be located at diverse locations within a cell. Note that a multicast group may span over multiple cells, however, since multicast scheduling applies to one base station, we restrict our consideration only to one cell. The primary difficulty in multicast scheduling for 3G cellular data networks stems from the mismatch in data rates attainable by individual users within a multicast group. Recall that if the base station transmits data at a higher rate than the maximum rate that a user's mobile device can handle, then the device is incapable of decoding *any* of the transmitted data. Since all users in a multicast group must be subject to the same transmission rate picked by the base station at each time slot, it is difficult to find one rate at which the base station sends multicast to a group. If it sends at the highest rate that users ask for, then there will be many users who may not get the transmission, and if the base station sends at the lowest rate requested by the users in a group, then other users with higher DRC rates (better channel condition) will be subject to rates lower than their DRCs. Therefore the challenging job of the multicast base station is that at each time slot, the base station must decide *which group* to transmit to, and choose *what rate* to transmit to that group as shown in Figure 1.

One simple way of multicast scheduling is to fix the transmission rate to a default value and do a round-robin among all the groups. The default rate is typically set to handle the DRC values of potential users located at the edge of a cell. This rate is the worst rate because it assumes that there is always a user at the edge of the cell regardless whether such a user is actually present or not. The current CDMA2000 1xEV-DO networks use this approach with a fixed rate of 204.8 kbps [6]. However, this scheme does not take into consideration user DRC rates, so it significantly limits the throughput of users, especially for those close to the base station with good channel conditions. Furthermore, this scheme does not necessarily maximize any form of user utility, therefore it is oblivious towards fairness between users and between groups.

Perhaps the natural solution to improve multicast data throughput is to partition users with similar channel conditions into the same multicast group. However, because of the channel condition dynamics and significant signaling overhead associated with group membership changes, such a solution is not practical. Our goal is to improve multicast throughput without imposing membership changes. Instead of fixing multicast data rates to the lowest rate requested by users in a group, we propose two multicast scheduling schemes that leverage the unicast feedback of user DRC rates to select an appropriate multicast data rate and a group for transmission in each TDM slot. To improve the system throughput, our algorithms may select higher rates than the lowest rate requested by a group. This means that when the base station chooses a multicast group and its transmission rate, some users in that group may not be able to receive the transmitted data. Therefore choosing the group data rate is as important as deciding the group to transmit to. Because user channel conditions vary, different users will miss different packets in the multicast stream. We next describe how such a scheduler is useful in different application scenarios.

In this paper, we consider two different types of multicast application scenarios and present two multicast scheduling algorithms that maximize two different utility functions. The first is applicable to delay tolerant cooperative data downloads while the second applies to multimedia content distribution for typical 3G multicast data networks. In the first scenario, the objective of this scheduler is to maximize the sum of $\log T_k^g$ for all groups. We assume that for group k, the utility of the entire group is $\log T_k^g$, where T_k^g is the group throughput for group k and the group throughput is defined as the sum of individual user receiving throughput within a group. We call this scheduler the *Inter-group Proportional Fair* (IPF) scheduler. IPF is likely to be useful in delay tolerant networks, possibly with nomadic users who have intermittent connectivity. IPF is useful when group members can cooperatively download data (which they share within the group), perhaps by forming an ad-hoc network. The original data could be source coded, e.g. using digital fountain type codes [14], [15], and the downloaded data could be subsequently reconciled within a group [16]. In the second scenario, multiple groups of users stay in a cell and these users receive some (possibly different) multimedia content from the base station. We consider a utility function of the form, $\log T_i$, where T_i is the receiving throughput of user i. User i's utility increases if more packets are received. The scheduler's objective is to maximize the sum of $\log T_i$ for all user i's in a cell; we call such a scheduler *Multicast Proportional Fair* (MPF) scheduler as it needs to achieve proportional fairness [17] (because of the log utility function) within the constraint of multicast. Recall that the MPF scheduler will sometimes select data rates that cause certain users (at the far edge of the cell) not to be able to receive data during certain time slots. This will certainly translate into lost packets and subsequently lost application frames for the users. Ideally, while users with poor channel conditions receive a "base" level of service, users with better channel conditions receive a higher quality data service.

The rest of the paper is organized as follows. In Section II we introduce our system model; in Sections III and IV, we present our IPF and MPF algorithms and provide proofs of the proportional fairness property; in Section V, we describe our simulation setup and results; in Section VI, we present an overview of related work. We conclude in Section VII.

II. Model Description

We consider a system with one base station (BS), and only multicast transmission is scheduled. Mobile devices are capable of maintaining unicast and multicast transmissions simultaneously and the base station uses the DRC feedback for the unicast connection to determine multicast data rate. Throughout this paper, we use the terms user, terminal, and mobile device/access terminal (AT) interchangeably.

We use the following notation:

- G: the number of multicast groups.
- $r_{ik}(t)$: DRC of Access Terminal (AT) i in group k at time t. We assume $0 < r_{min} \leq r_{ik}(t) \leq r_{max}$ where r_{min} and r_{max} represent the minimum and maximum possible DRC, respectively, and set $D = r_{max} - r_{min} + 1$.
- $r^g_k(t)$: feasible rate assigned to group k at time t.
- $T_{ik}(t)$: the (exponential/moving) average throughput of AT i in group k at time t.
- S_k: size of group k, i.e., total number of ATs in group k.

- $\vec{r}_i(t) = \{r_{1i}(t), ..., r_{S_i i}(t)\}$: DRC of group i at time t.
- $X = \{\vec{r}_1(t), \vec{r}_2(t), ..., \vec{r}_G(t)\} \in \Omega$ is a DRC vector of the system where Ω is a collection of all feasible DRC vectors. We say X is feasible, i.e., $X \in \Omega$ if each component of X lies between r_{min} and r_{max}. Let P be the number of all feasible DRC vectors, i.e., $\Omega = \{X_1, X_2, \ldots, X_P\}$ where $X_i = \{x^i_{1,1}, x^i_{2,1}, \ldots, x^i_{S_1,1}, x^i_{1,2}, \ldots, x^i_{S_G,G}\}$. We assume $x^i_{m,k}$ $(i = 1, 2, \ldots, P;\ k = 1, 2, \ldots, G;\ m = 1, 2, \ldots, S_k)$ are all positive integers, as well as r_{min}, r_{max} (or else we can rescale them) and define

$$R_i = r_{min} + i - 1. \quad (i = 1, 2, \ldots, D) \tag{1}$$

Let $\mathcal{S}$ denote the scheduler under discussion and

$$1^{\mathcal{S}}_k(t) := \begin{cases} 1, & \mathcal{S} \text{ chooses to serve group } k \text{ at time } t \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Then, the (exponential/moving) average throughput is updated at each time t by $T_{ik}(t+1) = (1-\frac{1}{t_c})T_{ik}(t) + \frac{1}{t_c} r^g_k(t) 1^s_k(t) 1_{\{r^g_k(t) \leq r_{ik}(t)\}}$ where t_c is latency time scale in number of time slots. In other words, AT i in group k will receive rate of $r^g_k(t)$ only if the group k is selected and its DRC value $r_{ik}(t)$ is no less than $r^g_k(t)$. Similarly, we define

$$I^{\mathcal{S}}_i(t) := \begin{cases} 1, & \mathcal{S} \text{ chooses to serve at rate } R_i \text{ at time } t \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

III. Inter-group PF Scheduler

In this section we propose a multicast scheduler that achieves inter-group proportional fairness (IPF). Denote the aggregate throughput of group k at time t by $T^g_k(t)$ and the aggregate rate of all ATs in group k at time t when the BS's transmission rate is y by $\phi_{k,t}(y)$, where

$$T^g_k(t) := \sum_{i=1}^{S_k} T_{ik}(t). \tag{4}$$

$$\phi_{k,t}(y) = \sum_{n=1}^{S_k} y 1_{\{y \leq r_{nk}(t)\}}. \tag{5}$$

Let $T^{\mathcal{S}}_k$ be the long-term arithmetic average throughput of group k under scheduler $\mathcal{S}$. Then, we have

$$T^{\mathcal{S}}_k = \lim_{N \to \infty} \frac{1}{N} \sum_{t=1}^{N} 1^{\mathcal{S}}_k(t) \phi_{k,t}(r^g_k(t)), \tag{6}$$

and similarly for $T_k^{\mathcal{S}^*}$ with $1_k^{\mathcal{S}}(t)$ replaced by $1_k^{\mathcal{S}^*}(t)$. We assume that the long-term average throughput always exists [12] for any multicast scheduler $\mathcal{S}$ under our consideration. In this case, note that we can also write $T_k^{\mathcal{S}} = \lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} T_k^g(t)$, i.e., the arithmetic time average and exponential average become identical in the limit.

Definition 1: A scheduler $\mathcal{S}^*$ is IPF if, for any other scheduler $\mathcal{S}$,

$$\sum_{k=1}^{G} \frac{T_k^{\mathcal{S}} - T_k^{\mathcal{S}^*}}{T_k^{\mathcal{S}^*}} \leq 0, \tag{7}$$

where $T_k^{\mathcal{S}}$ and $T_k^{\mathcal{S}^*}$ are defined as in (6). In other words, the aggregate of proportional changes in the long-term average group throughput of $\mathcal{S}^*$ caused by any other scheduler $\mathcal{S}$ must be non-positive. □

A. IPF Scheduling Algorithm

IPF scheduler $\mathcal{S}^*$: The feasible rate assigned to group k at time t is

$$r_k^g(t) = \arg\max_{y} \phi_{k,t}(y). \tag{8}$$

The BS chooses group $k(t)$ to transmit at rate $r_k^g(t)$,

$$\text{where } k(t) = \arg\max_{1\leq k\leq G} \frac{\phi_{k,t}(r_k^g(t))}{T_k^g(t)}. \tag{9}$$

Note that $r_k^g(t)$ represents the transmission rate at which the aggregate rate of group k is maximized. We observe that $r_k^g(t)$ is always equal to the DRC of some AT in that group. To see this, suppose otherwise that $r_k^g(t)$ falls between DRC values of two ATs in that group. Then, from (5), we can always increase $r_k^g(t)$ to the larger DRC (between these two values), which further increases $\phi_{k,t}$, and this leads to a contradiction. In consequence, we call $r_k^g(t)$ the *DRC of group* k, which plays the similar role in selecting a group to transmit, as the DRC of one node does in selecting a node to transmit in the unicast setting [9], [13], [18].

Once the *DRC of each group* is determined, the BS selects group $k(t)$ as indicated by (9). Note that this is similar to what unicast PF scheduler does, i.e., preferring to the group that receives smaller amount of service ($T_k^g(t)$) up till now compared to its capability ($\phi_{k,t}(r_k^g(t))$) [18]. Note that the definition of IPF in (7) is ill-defined when $T_h^{\mathcal{S}^*} = 0$ for some

h. Under the proposed scheduler $\mathcal{S}^*$, however, we show that this never happens (see [19]).

B. Inter-group PF

We here prove that scheduler $\mathcal{S}^*$ achieves IPF. To proceed, we define by $f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)$ $(1 \le i, j \le D;\ 1 \le k, h \le G;\ 1 \le p \le P)$ the empirical probability that the DRC vector is X_p, $\mathcal{S}^*$ selects group k, transmission rate R_i as defined in (1), and $\mathcal{S}$ selects group h, transmission rate R_j. To be precise,

$$f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) \;=\; \lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} 1_k^{\mathcal{S}^*}(t) 1_h^{\mathcal{S}}(t) I_i^{\mathcal{S}^*}(t) I_j^{\mathcal{S}}(t) \times 1_{\left\{\{r_{11}(t),\dots,r_{S_G G}(t)\}=X_p\right\}} \tag{10}$$

where $1_k^{\mathcal{S}^*}$ and $1_h^{\mathcal{S}}$ are defined in (2), whereas $I_i^{\mathcal{S}^*}$ and $I_j^{\mathcal{S}}$ are defined in (3). $f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)$ can be looked as the joint probability mass function (pmf) indicating the probability that {group index, AT index} given by $\mathcal{S}^*$ and $\mathcal{S}$ are $\{k, i\}$ and $\{h, j\}$ respectively, when the DRC vector is X_p. From above arguments, we have the following:

Lemma 1: For any k, h, i, j, p, we have

$$\sum_{n=1}^{S_k} \frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) R_i 1_{\{x_{n,k}^p \ge R_i\}}}{T_k^{\mathcal{S}^*}} \ge \sum_{n=1}^{S_h} \frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) R_j 1_{\{x_{n,h}^p \ge R_j\}}}{T_h^{\mathcal{S}^*}}. \tag{11}$$

Proof: The result becomes trivial when $f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) = 0$. So, we only need to consider the case of $f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) > 0$. In this case, there exists t_m $(1 \le m \le T_f)$, where T_f is the total number of time slots, such that $1_k^{\mathcal{S}^*}(t_m) 1_h^{\mathcal{S}}(t_m) I_i^{\mathcal{S}^*}(t_m) I_j^{\mathcal{S}}(t_m) 1_{\left\{\{r_{11}(t_m),\dots,r_{S_G G}(t_m)\}=X^p\right\}} = 1$. Hence, all terms on the LHS of the above equation are equal to 1, and thus $r_k^g(t_m) = R_i$ for all such m since $I_i^{\mathcal{S}^*}(t_m) = 1$, which then leads to $\phi_{k,t}(r_k^g(t_m)) = \phi_{k,t}(R_i)$. Also, group k is selected by our scheduler $\mathcal{S}^*$ at every such t_m since $1_k^{\mathcal{S}^*}(t_m) = 1$. Consequently, from (9), we have

$$\frac{\phi_{h,t_m}(r_h^g(t_m))}{T_h^{\mathcal{S}^*}} \le \frac{\phi_{k,t_m}(r_k^g(t_m))}{T_k^{\mathcal{S}^*}}, \quad (1 \le m \le T_f). \tag{12}$$

When $h = k$, from (8), the result follows by noting that

$$\frac{\text{LHS of (11)}}{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)} = \frac{\phi_{k,t_m}(r_k^g(t_m))}{T_k^{\mathcal{S}^*}} \ge \frac{\phi_{k,t_m}(R_j)}{T_k^{\mathcal{S}^*}} = \frac{\text{RHS of (11)}}{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)}.$$

When $h \ne k$, we have

$$\frac{\text{LHS of (11)}}{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)} = \frac{\phi_{k,t_m}(r_k^g(t_m))}{T_k^{\mathcal{S}^*}} \ge \frac{\phi_{h,t_m}(r_h^g(t_m))}{T_h^{\mathcal{S}^*}} \tag{13}$$

$$\ge \frac{\phi_{h,t_m}(R_j)}{T_h^{\mathcal{S}^*}} = \frac{\text{RHS of (11)}}{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)}, \tag{14}$$

where the inequality in (13) is from (12) and (14) is from (8). This completes the proof. ■

Proposition 1: The multicast scheduler $\mathcal{S}^*$ described in (8) and (9) is IPF. □

Proof: By expanding $T_k^{\mathcal{S}^*}$ and $T_h^{\mathcal{S}}$ in terms of their joint pmf with respect to the DRC vector X_p, we have

$$T_k^{\mathcal{S}^*}=\sum_{m=1}^{S_k}\sum_{h=1}^{G}\sum_{p\in\mathbb{P},i,j\in\mathbb{D}} f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)R_i 1_{\{x_{m,k}^p\geq R_i\}},\ T_h^{\mathcal{S}}=\sum_{n=1}^{S_h}\sum_{k=1}^{G}\sum_{p\in\mathbb{P},i,j\in\mathbb{D}} f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)R_j 1_{\{x_{n,h}^p\geq R_j\}}, \tag{15}$$

where $\mathbb{P}=\{1,...P\}$ and $\mathbb{D}=\{1,...D\}$.

For notational simplicity, we use $\sum_{p,i,j}$ instead of $\sum_{p\in\mathbb{P}}\sum_{i,j\in\mathbb{D}}$. Observe now that

$$\begin{aligned}\sum_{h=1}^{G}\frac{T_h^{\mathcal{S}}}{T_h^{\mathcal{S}^*}} &= \sum_{h=1}^{G}\sum_{n=1}^{S_h}\sum_{k=1}^{G}\sum_{p,i,j}\frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)R_j 1_{\{x_{n,h}^p\geq R_j\}}}{T_h^{\mathcal{S}^*}}=\sum_{h=1}^{G}\sum_{k=1}^{G}\sum_{p,i,j}\sum_{n=1}^{S_h}\frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)R_j 1_{\{x_{n,h}^p\geq R_j\}}}{T_h^{\mathcal{S}^*}}\\ &\leq \sum_{h=1}^{G}\sum_{k=1}^{G}\sum_{p,i,j}\sum_{n=1}^{S_k}\frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)R_i 1_{\{x_{n,k}^p\geq R_i\}}}{T_k^{\mathcal{S}^*}}=\sum_{k=1}^{G}\sum_{n=1}^{S_k}\sum_{h=1}^{G}\sum_{p,i,j}\frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)R_i 1_{\{x_{n,k}^p\geq R_i\}}}{T_k^{\mathcal{S}^*}}\\ &= \sum_{k=1}^{G}\frac{T_k^{\mathcal{S}^*}}{T_k^{\mathcal{S}^*}}=\sum_{h=1}^{G}\frac{T_h^{\mathcal{S}^*}}{T_h^{\mathcal{S}^*}},\end{aligned}$$

where both the first and the fourth equalities are from (15) and the inequality follows from Lemma 1. This shows (7) and we are done. ■

In our IPF algorithm, each group has to decide its "group DRC" r^g as a representative value [1] (See (8)). Depending on the DRC values of users in that group ($r_i,\ i=1,2,\ldots,S$), some of them will get that rate and some of them will not (zero rate). Note that the BS does not care why each group asks for that rate; it only performs the usual PF scheduling based on these group DRCs. We already saw that under IPF, the group rate r^g maximizes the total aggregate rate to that group if the BS chooses that group for service at rate r^g.

As briefly mentioned in the introduction, we can interpret the IPF algorithm as a fair and reliable transfer in delay-tolerant, lossy multicast networks. Consider a multicast network with some delay-tolerant applications where each packet transmission is subject to high error/loss due to severe wireless link characteristics or interference, etc. Let q be the probability that each packet transmission (from the BS to each user in a chosen group) is successful. We assume q is very small (unreliable network) and that each packet is subject to an error independently of everything else. Note that under IPF scheduler $\mathcal{S}^*$, we have

$$r^g=\arg\max_y \phi(y), where\ \ \phi(y)=\sum_{i=1}^{S} y 1_{\{y\leq r_i\}}. \tag{16}$$

[1] For simple exposition, we here suppress the group index k and time index t from the original definitions.

Suppose the BS chooses rate y for this group (y packets per unit time slot). Then, the number of users $M = M(y)$ that receive rate of y (non-zero rate) in that group becomes $M = \sum_{i=1}^{S} 1_{\{y \leq r_i\}}$. Note that each packet transmission will be successful with probability q, and that there are M such duplicate packets. Thus, the probability P that at least one of these M packets will get through (delivered to at least one user in that group) becomes $P = 1-(1-q)^M \approx qM$ for small q. Then, the average number of successful packets $\mathbb{E}\{N_s\}$ delivered to that group becomes

$$\mathbb{E}\{N_s\} = yP \approx qyM = q\sum_{i=1}^{S} y 1_{\{y\leq r_i\}} = q\phi(y). \tag{17}$$

So $\mathbb{E}\{N_s\}$ is maximized when y is the maximizer of the RHS of (17). Note that, this is exactly the group rate selection algorithm for IPF as in (16). In other words, under IPF, the group DRC corresponds to the "most reliable" rate for that group.

C. General Inter-group PF Scheduler

In this section, we extend proportional fairness to be more general. We use the notion of $(\vec{p}, \alpha)$ PF [11] to develop more candidate schedulers which have different weights in tradeoff between throughput and fairness. While the $(\vec{p}, \alpha)$ PF ($\vec{p} = (p_1, ...p_m)$ where $p_i > 0$) can be achived in unicast setting [12], in multicast, it is only possible when $p_1 = p_2 = ...p_m$. In other words, we can design multicast schedulers that is $(\vec{e}, \alpha)$ PF.

Definition 2: A scheduler $\mathcal{S}^*(\alpha)$ is $(\vec{e}, \alpha)$ IPF if, for any other scheduler $\mathcal{S}$,

$$\sum_{h=1}^{G} \frac{T_h^{\mathcal{S}} - T_h^{\mathcal{S}^*(\alpha)}}{(T_h^{\mathcal{S}^*(\alpha)})^{\alpha}} \leq 0, \text{ where } \vec{e} = \{1, 1, ...1\} \text{ and } 0 \leq \alpha \leq 1. \tag{18}$$

We propose the following set of schedulers:

$(\vec{e}, \alpha)$ ***IPF scheduler*** $\mathcal{S}^*(\alpha)$**:** The feasible rate assigned to group k is $r_k^g(t)$ defined in (8) (same as scheduler $\mathcal{S}^*$). The BS chooses group $k(t)$ to transmit at rate $r_k^g(t)$ where

$$k(t) = \arg \max_{1\leq k\leq G} \frac{\phi_{k,t}(r_k^g(t))}{(T_k^g(t))^{\alpha}}. \tag{19}$$

Here, $T_k^g(t)$, $\phi_{k,t}(y)$ are defined in (4), (5), respectively.

Similarly as in the proof of Proposition 1, we can show that $\mathcal{S}^*(\alpha)$ is $(\vec{e}, \alpha)$ IPF.

IV. Multicast PF Scheduler

In this section we propose a multicast scheduler that achieves the proportional fairness among all the users in the system. To distinguish it from PF in the unicast setting, we call it Multicast PF (MPF). Let $T^{\mathcal{S}}_{m,k}$ be the long-term arithmetic average throughput of AT m in group k under scheduler $\mathcal{S}$. Then, we have

$$T^{\mathcal{S}}_{m,k} = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} 1^{\mathcal{S}}_{k}(t) r^{g}_{k}(t) 1_{\{r^{g}_{k}(t) \le r_{mk}(t)\}}. \tag{20}$$

As in Section III, we assume that the long-term average throughput exists for any multicast scheduler $\mathcal{S}$ under consideration.

Definition 3: A scheduler $\mathcal{S}^{\diamond}$ is MPF if, for any other scheduler $\mathcal{S}$,

$$\sum_{h=1}^{G} \sum_{n=1}^{S_h} \frac{T^{\mathcal{S}}_{n,h} - T^{\mathcal{S}^{\diamond}}_{n,h}}{T^{\mathcal{S}^{\diamond}}_{n,h}} \le 0, \tag{21}$$

where $T^{S}_{n,h}$ and $T^{\mathcal{S}^{\diamond}}_{n,h}$ are defined as in (20). In other words, the aggregate of proportional changes in the long-term average user throughput of $\mathcal{S}^{\diamond}$ caused by any other scheduler $\mathcal{S}$ must be non-positive. □

A. MPF Scheduling Algorithm

MPF scheduler $\mathcal{S}^{\diamond}$***:*** The feasible rate assigned to group k at time t is

$$r^{g}_{k}(t) = \arg\max_{y} \varphi_{k,t}(y), \tag{22}$$

$$\text{where } \varphi_{k,t}(y) = \sum_{n=1}^{S_k} \frac{y}{T_{nk}(t)} 1_{\{y \le r_{nk}(t)\}}. \tag{23}$$

The BS chooses group $k(t)$ to transmit at rate $r^{g}_{k}(t)$

$$\text{where } k(t) = \arg\max_{1\le k\le G} \varphi_{k,t}(r^{g}_{k}(t)). \tag{24}$$

Note that $r^{g}_{k}(t)$ is the transmission rate at which 'weighted' aggregate rate of group k is maximized, where the weight of each AT is $1/T_{nk}(t)$, and $T_{nk}(t)$ is the (exponential/moving) average throughput of AT n in group k at time t. Thus, if the AT has received less throughput so far (smaller T_{nk}), then that AT will have more weight (importance) in determining the feasible rate $r^{g}_{k}(t)$ in (22) as well as the group selection in (24), since smaller T_{nk} will make

$\varphi_{k,t}(r_k^g(t))$ also larger. As a special case, when there is only one user in each group, i.e., the size of all groups is 1, note that MPF scheduler $\mathcal{S}^\diamond$ becomes the usual unicast PF scheduler [9], [13], [18]. Under the proposed scheduler $\mathcal{S}^\diamond$, we can also show $T_{m,k}^{\mathcal{S}^\diamond} > 0$ (see [19] for the detailed proof).

B. Multicast PF

Define $f_{k,h,i,j}^{\mathcal{S}^\diamond,\mathcal{S}}(p)$ in the same way as in (10) with $\mathcal{S}^*$ replaced by $\mathcal{S}^\diamond$. The following will then be used to prove MPF:

Lemma 2: For any k, h, i, j, p, we have

$$\sum_{n=1}^{S_k} \frac{f_{k,h,i,j}^{\mathcal{S}^\diamond,\mathcal{S}}(p) R_i 1_{\{x_{n,k}^p \geq R_i\}}}{T_{n,k}^{\mathcal{S}^\diamond}} \geq \sum_{n=1}^{S_h} \frac{f_{k,h,i,j}^{\mathcal{S}^\diamond,\mathcal{S}}(p) R_j 1_{\{x_{n,h}^p \geq R_j\}}}{T_{n,h}^{\mathcal{S}^\diamond}}. \tag{25}$$

Proof: The proof follows the same line as in Lemma 1 by using (22) – (24) instead of (8) and (9). The result becomes trivial when $f_{k,h,i,j}^{\mathcal{S}^\diamond,\mathcal{S}}(p) = 0$. So, we only need to consider the case of $f_{k,h,i,j}^{\mathcal{S}^\diamond,\mathcal{S}}(p) > 0$. In this case, there exists t_m $(1 \leq m \leq T_f)$ such that $1_k^{\mathcal{S}^\diamond}(t_m) 1_h^{\mathcal{S}}(t_m) I_i^{\mathcal{S}^\diamond}(t_m) I_j^{\mathcal{S}}(t_m) 1_{\left\{\{r_{11}(t_m),\ldots,r_{S_G G}(t_m)\}=X^p\right\}} = 1$. Hence, all terms on the LHS of the above equation are equal to 1, and thus $r_k^g(t_m) = R_i$ for all such m since $I_i^{\mathcal{S}^\diamond}(t_m) = 1$, which then leads to $\varphi_{k,t}(r_k^g(t_m)) = \varphi_{k,t}(R_i)$. Also, group k is selected by our scheduler $\mathcal{S}^\diamond$ at every such t_m since $1_k^{\mathcal{S}^*}(t_m) = 1$. Consequently, from (24), we have

$$\varphi_{h,t_m}(r_h^g(t_m)) \leq \varphi_{k,t_m}(r_k^g(t_m)), \quad (1 \leq m \leq T_f). \tag{26}$$

When $h = k$, from (22) and (23) , the result follows by noting that

$$\frac{\text{LHS of (25)}}{f_{k,h,i,j}^{\mathcal{S}^\diamond,\mathcal{S}}(p)} = \varphi_{k,t_m}(r_k^g(t_m)) \geq \varphi_{k,t_m}(R_j) = \frac{\text{RHS of (25)}}{f_{k,h,i,j}^{\mathcal{S}^\diamond,\mathcal{S}}(p)}.$$

When $h \neq k$, we have

$$\frac{\text{LHS of (25)}}{f_{k,h,i,j}^{\mathcal{S}^\diamond,\mathcal{S}}(p)} = \varphi_{k,t_m}(r_k^g(t_m)) \geq \varphi_{h,t_m}(r_h^g(t_m)) \tag{27}$$

$$\geq \varphi_{h,t_m}(R_j) = \frac{\text{RHS of (25)}}{f_{k,h,i,j}^{\mathcal{S}^\diamond,\mathcal{S}}(p)}, \tag{28}$$

where the inequality in (27) is from (26) and (28) is from (22). This completes the proof. ■

Proposition 2: The multicast scheduler $\mathcal{S}^\diamond$ described in (22)–(24) is MPF. □

Proof: Expand $T^{\mathcal{S}^\diamond}_{m,k}, T^{\mathcal{S}}_{n,h}$ in terms of their joint pmf with respect to the DRC vector X_p,

$$T^{\mathcal{S}^\diamond}_{m,k} = \sum_{h=1}^{G} \sum_{p\in\mathbb{P},i,j\in\mathbb{D}} f^{\mathcal{S}^\diamond,\mathcal{S}}_{k,h,i,j}(p) R_i 1_{\{x^p_{m,k}\geq R_i\}}, \quad T^{\mathcal{S}}_{n,h} = \sum_{k=1}^{G} \sum_{p\in\mathbb{P},i,j\in\mathbb{D}} f^{\mathcal{S}^\diamond,\mathcal{S}}_{k,h,i,j}(p) R_j 1_{\{x^p_{n,h}\geq R_j\}}, \tag{29}$$

where $\mathbb{P} = \{1, ...P\}$ and $\mathbb{D} = \{1, ...D\}$. Then the proof becomes similar to that of Proposition 1, by using (29) and Lemma 2. For notational simplicity, we use $\sum_{p,i,j}$ instead of $\sum_{p\in\mathbb{P}}\sum_{i,j\in\mathbb{D}}$. Observe now that

$$\begin{aligned}
\sum_{h=1}^{G}\sum_{n=1}^{S_h} \frac{T^{\mathcal{S}}_{n,h}}{T^{\mathcal{S}^\diamond}_{n,h}} &= \sum_{h=1}^{G}\sum_{n=1}^{S_h}\sum_{k=1}^{G}\sum_{p,i,j} \frac{f^{\mathcal{S}^\diamond,\mathcal{S}}_{k,h,i,j}(p) R_j 1_{\{x^p_{n,h}\geq R_j\}}}{T^{\mathcal{S}^\diamond}_{n,h}} = \sum_{h=1}^{G}\sum_{k=1}^{G}\sum_{p,i,j}\sum_{n=1}^{S_h} \frac{f^{\mathcal{S}^\diamond,\mathcal{S}}_{k,h,i,j}(p) R_j 1_{\{x^p_{n,h}\geq R_j\}}}{T^{\mathcal{S}^\diamond}_{n,h}} \\
&\leq \sum_{h=1}^{G}\sum_{k=1}^{G}\sum_{p,i,j}\sum_{n=1}^{S_k} \frac{f^{\mathcal{S}^\diamond,\mathcal{S}}_{k,h,i,j}(p) R_i 1_{\{x^p_{n,k}\geq R_i\}}}{T^{\mathcal{S}^\diamond}_{n,k}} = \sum_{h=1}^{G}\sum_{n=1}^{S_k}\sum_{k=1}^{G}\sum_{p,i,j} \frac{f^{\mathcal{S}^\diamond,\mathcal{S}}_{k,h,i,j}(p) R_i 1_{\{x^p_{n,k}\geq R_i\}}}{T^{\mathcal{S}^\diamond}_{n,k}} \\
&= \sum_{k=1}^{G}\sum_{n=1}^{S_k} \frac{T^{\mathcal{S}^\diamond}_{n,k}}{T^{\mathcal{S}^\diamond}_{n,k}} = \sum_{h=1}^{G}\sum_{n=1}^{S_h} \frac{T^{\mathcal{S}^\diamond}_{n,h}}{T^{\mathcal{S}^\diamond}_{n,h}}
\end{aligned}$$

where both the first and the fourth equalities are from (29) and the inequality follows from Lemma 2. This shows (21) and we are done. ■

C. General MPF Scheduler

In this section, we extend the notion of MPF to the general notion of $(\vec{e}, \alpha)$ MPF.

Definition 4: A scheduler $\mathcal{S}^\diamond(\alpha)$ is $(\vec{e}, \alpha)$ MPF if, for any other scheduler $\mathcal{S}$,

$$\sum_{h=1}^{G}\sum_{n=1}^{S_h} \frac{T^{\mathcal{S}}_{n,h} - T^{\mathcal{S}^\diamond(\alpha)}_{n,h}}{(T^{\mathcal{S}^\diamond(\alpha)}_{n,h})^\alpha} \leq 0, \;\; \textit{where} \;\; \vec{e} = \{1, 1, ...1\}, \; 0 \leq \alpha \leq 1. \tag{30}$$

We propose the following set of schedulers:

$(\vec{e}, \alpha)$ ***MPF scheduler*** $\mathcal{S}^\diamond(\alpha)$***:*** The feasible rate assigned to group k at time t is

$$r^g_k(t) = \arg\max_y \varphi^\alpha_{k,t}(y), \tag{31}$$

$$\textit{where } \varphi^\alpha_{k,t}(y) = \sum_{n=1}^{S_k} \frac{y}{(T_{nk}(t))^\alpha} 1_{\{y\leq r_{nk}(t)\}}. \tag{32}$$

The BS chooses group $k(t)$ from (24) to transmit at rate $r^g_{k(t)}(t)$.

Following the same line as in Proposition 2, we can show that $\mathcal{S}^\diamond(\alpha)$ is $(\vec{e}, \alpha)$ MPF.

V. Simulations

To evaluate the performance of our algorithms, we conduct packet level simulation using ns-2 [20]. We consider one base station serving multiple mobile users with random locations.

We generate a DRC trace for each user as follows: at each time slot t, the DRC value for user i is predicted based on its position and a simulated channel fading process considering both slow fading and fast fading. Channel is modeled with slow fading as a function of the client's distance from the base station, and fast fading using Rayleigh fading. The combined effect is then mapped to a list of supported DRC values (in kbps) of $\{0, 38.4, 76.8, 153.6, 204.8, 307.2, 409.6, 614.4, 921.6, 1228.8, 1843.2, 2457.6\}$ according to CDMA2000 1xEV-DO specification. We assume the input buffer at the base station is constantly backlogged. We run the simulation for 30 seconds, and evaluate the performance of different algorithms listed in Table I with different group formation using the same DRC traces. In order to avoid transient effects, we discard results from the initial 3 seconds.

Table I shows the four scheduling algorithms we test in our simulation studies. Among them, IPF and MPF have been discussed in Sections III and IV respectively. Recall that we provide IPF and MPF in multicast systems for tradeoff between throughput and fairness. The best way to test their performance is to compare them with other algorithms that achieve the maximum throughput or optimal fairness. This is why MAX and MIN schedulers are provided. MAX aims at the maximum aggregate throughput, and MIN gives every user in the same group the absolutely fair share.

A. Objective Functions

As shown in Section III and IV, different algorithms maximize different objective functions. Given that T_i is user i's throughput and T_k^g is group k's throughput, MAX maximizes the aggregate throughput of all users, $\sum_i T_i$, MPF maximizes the sum of log utility function of individual user throughput, $\sum_i \log T_i$, which is a measure of total happiness of individual users, and IPF maximizes the sum of log utility function of group throughput $\sum_k \log T_k^g$ and group throughput T_k^g is measured by the aggregate of user throughput in a group k. In the case of IPF, the objective is to achieve the highest group happiness rather than individual happiness.

We conduct two sets of experiments to verify that these algorithms do indeed maximize their respective objective functions given various group configurations and to see the performance difference among various algorithms in terms of the given objective functions. In the first set of experiments, we fix the number of users served by the base station to 32, and divide them into equal-sized groups. Each setup can be represented by a tuple (n, g), where n is the number of groups and g is the group size. We conduct simulation runs for the following 6 values of (n, g): (1,32), (2,16), (4,8), (8,4), (16,2), (32,1). When there is only one group, MAX and IPF behave the same way as noted earlier and when the number of group is 32, this case degenerates to a unicast scenario. As a result, IPF, MPF, and MIN show exactly same result as in the unicast case. This setup allows us to examine the performance of schedulers under various network scenarios.

Figures 2, 3, and 4 plot the values of the three objective functions for the above experiment. Figure 2 shows the value of $\sum \log T_i$. As expected, MPF shows the highest value as it is designed to maximize that objective function. When there is only one group, since MAX and IPF behave the same way, their values are the same. The objective value decreases as the number of groups increases. This is because as the number of groups increases while fixing the number of total users, the system degenerates into a unicast scenario. Thus, the benefit of multicast diminishes, so the total throughput decreases (as shown in Figure 4). Under the unicast case, we see that IPF, MPF, and MIN behave the same way. As IPF is designed to achieve fairness among groups, it does not maximize this objective value and thus, shows smaller values than MPF. MIN tends to perform fairly well in terms of fairness as the utility value is pretty high since all group members are receiving data all the time, albeit, at the minimum rate, but as we see in Figure 4, its total throughput is very low.

Figure 3 shows the values of $\sum \log T_k^g$. This objective function measures the total log utility of group throughput that is the sum of member throughput in each group, and in some sense, it makes the optimal tradeoff between group throughput and fairness among groups. Since we are taking log on the group aggregate throughput, the actual magnitude of values is small and the difference between two values is also very small. But in actual values, we observe the difference is very large in Figure 4. According to this metric, we find that IPF performs the best. Under the unicast scenario, we can find that the values become the maximum. Moreover, note that $\sum \log T_k^g$ increases as the number of groups increase. To see this, consider this objective function, $\sum \log T_k^g = \log\left(T_1^g \times T_2^g \times ....T_G^g\right)$ where G is the number of groups. On one hand, the increasing of G increases the number of factors. On the other hand, as G increases, the throughput of each group, T_k^g $(1 \leq k \leq G)$, tends to decrease because of decrease of the number of users in each group. When $T_k^g >> 1$ regardless the number of users in a group, which is the case of our situation, the effect of increasing number of factors will dominate and as a result the product increases, so as $\sum \log T_k^g$. This coincides with our simulation result.

Figure 4 shows the total sum of user throughput. In this metric, MAX performs the best as expected. We also find that MAX performs the worst in Figures 2 and 3. This indicates that MAX is a very greedy scheme that although it achieves high total throughput in the system, it is unfair among users (as seen in Figure 2) and among groups (as seen in Figure 3). We observe that IPF and MPF give pretty good throughput. This result, along with the other results above, indicates that MPF and IPF strike a good balance between throughput and fairness.

B. Distribution of Throughput

In the above experiment, we have seen the average performance of the system after the system converges. In this section, we are interested in the distribution of throughput among users and groups under various schedulers. This sheds more light into the tradeoff between fairness and throughput that each scheme makes. As mentioned before, MIN gives each user in the same group the equal share. However, recall that the default scheduler FIXED used in CDMA2000 1xEV-DO systems gives equal share even to users not in the same group with the rate of 204.8 kbps [6]. Hence, in this experiment, we also take FIXED into consideration.

We conduct the following experiment. We divide 100 users randomly placed in a cell into 10 groups, each with 10 users. Then we run various schedulers and plot the average throughput of each user measured for each 3-second interval during the entire simulation run. Figure 5 shows the results of the experiment. We find that FIXED serves all users with a fixed rate, but penalizes both distant users and close users from the base station as throughput for distant users becomes zero while close users are subject to low throughput even if it can receive data at a higher rate. MIN maintains low throughput for all users, but no users will starve. MAX tends to favor only users close to the base station, so while close users have very high throughput, distant users receive nothing. IPF achieves higher throughput for close users than MPF because its rate selection algorithm always finds the rate that maximizes the aggregate group rate. But IPF tends to have larger variance in throughput than MPF and have a lower minimum than MPF. This is because MPF tries to provide fairness to distant users with poor channel states by giving them more chances to communicate than IPF does. In order for MPF to give more chances to distant users, it has to take some time slots from close users, which makes its maximum throughput lower than IPF.

From the same experiment run, we now plot the group throughput measured by the aggregate user throughput of each group. We plot them for all 10 groups in Figure 6. In the figure, we can find that groups 1, 4, 5, 6 and 10 have members very close to the base station (so we call them *close groups*) while the other groups tend to have more distant members (we call these groups *distant groups*). In every experiment we conduct, we choose user locations completely randomly. MAX shows the biggest skew of throughput distribution among groups as we find that some groups do not get any throughput while the close groups tend to get all the bandwidth. FIXED gives almost a constant bit rate to all groups (with rate variance due to user locations) and MIN has more varying group rates because it tries to achieve fairness while honoring only the worst user in each group. The difference between IPF and MPF is clearly shown in this figure. IPF tries to maximize the group rate while keeping the minimum group rate high. Thus, it tends to achieve very good balance between group throughput and fairness. MPF does not care much about the group throughput and focuses on equalizing user happiness. As the utility of users are equalized across groups, we tend to see less variance in the group rates, but it may not be able to achieve as high group throughput as IPF.

VI. Related Work

This paper proposes a suite of scheduling algorithms for multicast. While a large number of papers have proposed and evaluated unicast scheduling algorithms where multiple users share a time-varying wireless channel using Time Division Multiplexing (TDM) [6], [21], [7], [22], [2], [23], [8], [9], [24], [10], [11], [25], [12], [26], [13], to the best of our knowledge, no paper has addressed multicast scheduler design for cellular data networks using TDM such as CDMA2000 1xEV-DO.

The proportional fair scheduling algorithm for unicast is proposed in [9], [13], [26] for CDMA2000 1xEV-DO systems to maximize the log utility function. The weighted proportional fair schemes proposed in [22] demonstrate how one can choose a scheduler with an efficiency-fairness tradeoff between the two extreme cases, namely, a channel-unaware scheduler, and a channel-aware scheduler that serves the best mobile handset at any given time. The schemes proposed in [24] maximize liner utility functions, and they are based on stochastic approximation. Proportional fairness is generalized into $(p,\ \alpha)$-proportional fairness in [11] to unify the max-min fairness, proportional fairness and the worst case fairness, which is total-throughput maximization. A unified scheduler is proposed in [12] that achieves (p,α)-proportional fairness in terms of the asymptotic behavior of the long-term average throughput.

Properties of the rate region achievable by a general class of opportunistic scheduling algorithms are discussed in [23]. In addition, [23] also considers different fairness criteria and discusses optimal scheduling algorithms. The optimality of a general class of "gradient-like" opportunistic scheduling algorithms is proved in [21], [27], [28]. A scheduling algorithm that maintains mobile handset service rates in proportions to one another is proposed in [23]. The scheduling algorithm studied in [8] uses the assumption that each mobile handset has a finite amount of data to receive and it leaves the system once this data is received.

Broadcast scheduling algorithms are used in database and content distribution systems, taking into consideration application level information such as the size and importance of data items to be broadcast. In the context of 802.16e like mobile networks, where mobile hosts can go into sleep mode to save energy, [29] proposes a number of scheduling algorithms letting each host assign its own merit to each broadcast data item. The mechanisms divide each broadcast super-frame into a number of logical channels, and let mobile hosts decide which channel to listen to at the beginning of each super-frame in order to maximize the normalized throughput of the system for each super-frame. In the calculation for normalized throughput, each data item is multiplied by the merit for each host. Our work is quite different in the sense that we study the scheduler design for base stations in TDM systems without any application information.

VII. Conclusions

In this paper, we propose two sets of unified multicast proportional fair scheduling algorithms, the $(\vec{e}, \alpha)$ IPF schedulers and the $(\vec{e}, \alpha)$ MPF schedulers. Each of these schedulers supports a different utility function. In particular, the IPF scheduler supports the utility function of log of aggregate group throughput that is computed by summing up all the user throughput in a group, and the MPF scheduler supports the utility of log of individual user throughput. These multicast scheduling algorithms can be applied to different scenarios depending on the application and business model of ISPs. We compare the performance of these schedulers with several multicast scheduling algorithms and show through simulation that they achieve good balance between fairness and throughput of groups or users. We are investigating new scheduling algorithms to ensure QoS for multicast. One way to provide QoS is to ensure that the average transmission rate that each group or user gets does not fall below specified rates. We leave this as future work.

References

[1] "3g today," http://www.3gtoday.com/.

[2] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushayana, and A. Viterbi, "Cdma/hdr: A bandwidth-efficient high-speed wireless data service for nomadic users," *IEEE Communications Magazine*, vol. 38, no. 7, pp. 70–77, July 2000.
[3] 3GPP, "High speed downlink packet access (hsdpa): overall description," Mar 2003, http://www.3gpp.org.
[4] MobiTV, "Mobile television and radio service provider," http://www.mobitv.com.
[5] PC Magazine, "Verizon's vcast: Video over evdo phones," Jan 2005, http://www.pcmag.com/article2/0,1759,1749429,00.asp.
[6] P. Agashe, R. Rezaiifar, and P. Bender, "Cdma2000 high rate broadcast packet data air interface design," *IEEE Communications Magazine*, pp. 83–89, Feb 2004.
[7] R. Agrawal, A. Bedekar, R. La, and V. Subramanian, "A class and channel-condition based weighted proportionally fair scheduler for edge/gprs," in *ITCOM'01*, Denver, CO, August 2001.
[8] S. Borst, "User-level performance of channel-aware acheduling algorithms in wireless data networks," *IEEE/ACM Trans. Networking*, vol. 13, no. 3, pp. 636–647, 2005.
[9] A. Jalali, R. Padovani, and R. Pankaj, "Data throughput of CDMA-HDR a high efficiency-high data rate personal communication wireless system," in *Vehicular Technology Conference (VTC2000-Spring)*, Tokyo, Japan, May 2000, pp. 1854–1858.
[10] X. Liu, E. K. P. Chong, and N. B. Shroff, "A framework for opportunistic scheduling in wireless networks," *Computer Networks*, vol. 41, no. 4, pp. 451–474, 2003.
[11] J. Mo and Jean C. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Netw*, vol. 8, no. 5, pp. 556–567, 2000.
[12] Y. Ohta, M. Tsuru, and Y. Oie, "Fairness property and TCP-level performance of unified scheduling algorithm in HSDPA networks," in *Universal Multiservice Networks: Third European Conference, ECUMN 2004, Porto, Portugal, October 25-27, 2004. Proceedings.* 2004, vol. 3262, pp. 185–195, Springer.
[13] P. Viswanath, D. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," *IEEE Transactions on Information Theory*, vol. 48, no. 6, pp. 1277–1294, 2002.
[14] C. Gkantsidis and P. Rodriguez, "Network coding for large scale content distribution," in *INFOCOM'05*, 2005, Avalanche.
[15] J. W. Byers, M. G. Luby, M. Mitzenmacher, and A. Rege, "Digital fountain approach to reliable distribution of bulk data," in *SIGCOMM*, 1998, pp. 56–67.
[16] J. Byers, J. Considine, M. Mitzenmacher, and S. Rost, "Informed content delivery across adaptive overlay networks," in *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM)*, Pittsburgh, PA, USA, August 2002, pp. 47–60.
[17] F. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, vol. 8, pp. 33–37, Jan 1997.
[18] Y., M. Tsuru, and Y. Oie, "Fairness property and tcp-level performance of unified scheduling algorithm in hsdpa networks.," in *ECUMN*, 2004, pp. 185–195.
[19] H. Won, H. Cai, D. Eun, K. Guo, A. Netravali, , I. Rhee, and K. Sabnani, "Generalized dynamic rate multicast scheduling in third generation wireless data network," Tech. Rep., 2007, http://www4.ncsu.edu/∼hwon/paper/Techrep07-Multicast.pdf.
[20] "The network simulator- ns-2," http://www.isi.edu/nsnam/ns/.
[21] R. Agrawal and V. Subramania, "Optimality of certain channel aware scheduling policies," in *40th Annu. Allerton Conf. Communications, Control, and Computing*, Monticello, Illinois, USA, Oct 2002, pp. 1532–1541.
[22] R. Agrawal, A. Bedekar, R. La, and V. Subramanian, "A class and channel-condition based weighted proportionally fair scheduler," in *ITC'01*, Salvador, Brazil, December 2001.
[23] S. Borst and P. Whiting, "Dynamic rate control algorithms for HDR throughput optimization," in *IEEE Infocom'01*, Apr. 2001, pp. 976–985.
[24] X. Liu, E. K. P. Chong, and N. B. 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, 2001.
[25] Y. Ohta, M. Tsuru, and Y. Oie, "Framework for fair scheduling schemes in the next generation high-speed wireless links," in *the 8th International Conference on Cellular and Intelligent Communications (CIC 2003)*, Oct 2003.
[26] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*, Cambridge University Press, 2005.
[27] H. Kushner and P. Whiting, "Asymptotic properties of proportional fair sharing algorithms," in *40th Annu. Allerton Conf. Communications, Control, and Computing*, Monticello, Illinois, USA, Oct 2002.
[28] A. L. Stolyar, "On the asymptotic optimality of the gradient scheduling algorithm for multi-user throughput allocation," *Operations Research*, vol. 53, no. 1, pp. 12–25, 2005.
[29] R. Cohen and R. Rizzi, "On the trade-off between energy and multicast efficiency in 802.16e-like mobile networks," in *IEEE Infocom'06*, Barcelona, Spain, Apr 2006.

Algorithm	Rate selection scheme	Group selection scheme
IPF	$r_k^g(t) = \arg\max_y \phi_{k,t}(y)$	$k(t) = \arg\max_{1 \le k \le G} \frac{\phi_{k,t}(r_k^g(t))}{T_k^g(t)}$
MPF	$r_k^g(t) = \arg\max_y \varphi_{k,t}(y)$	$k(t) = \arg\max_{1 \le k \le G} \varphi_{k,t}(r_k^g(t))$
MAX	$r_k^g(t) = \arg\max_y \phi_{k,t}(y)$	$k(t) = \arg\max_{1 \le k \le G} \phi_{k,t}(r_k^g(t))$
MIN	$r_k^g(t) = \min_n r_{nk}(t)$	$k(t) = \arg\max_{1 \le k \le G} \frac{r_k^g(t) S_k}{T_k^g(t)}$

TABLE I

A SUMMARY OF SCHEDULING ALGORITHMS. THE DEFINITIONS FOR $\phi_{k,t}$, $\varphi_{k,t}$, AND $T_k^g(t)$ ARE DEFINED IN (5), (23) AND (4) RESPECTIVELY. n CAN BE ANY NUMBER IN $\{1, 2, \ldots, S_k\}$.

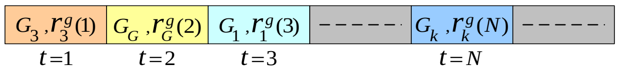

Fig. 1. Time slots for multicast schedulers. At each time slot, based on the DRC feedbacks from all users, one group is selected, and one data rate is selected to transmit to the chosen group. G_i represents the selected group i, and $r_i^g(t)$ represents transmission rate to group i at time slot t.

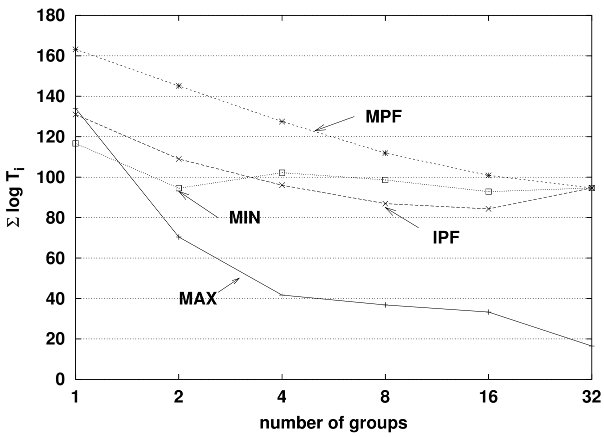

Fig. 2. The value of $\sum \log T_i$ for various multicast schedulers as we increases the number of groups from 1 to 32 while fixing the total number of users in a cell to 32. T_i is the throughput of user i. MPF is optimized for this metric.

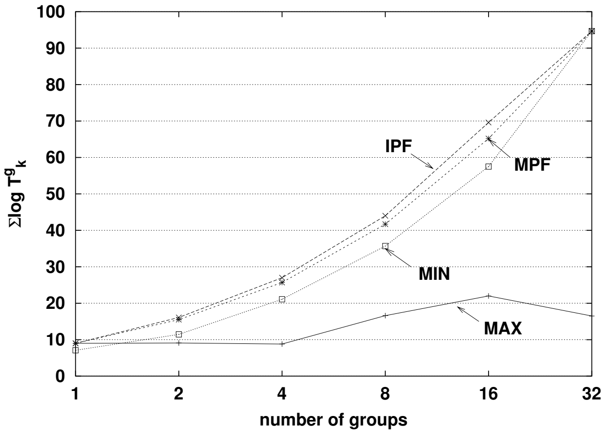

Fig. 3. The value of $\sum \log T_k^g$ for various multicast schedulers as we increases the number of groups from 1 to 32 while fixing the total number of users in a cell to 32. T_k^g is the total throughput of group k. IPF is optimized for this metric.

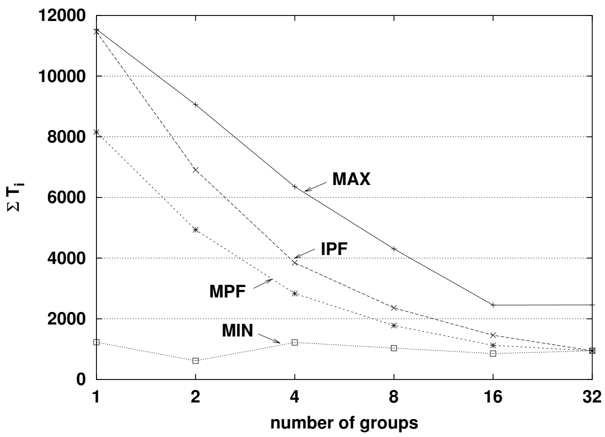

Fig. 4. The value of $\sum T_i$ for various multicast schedulers as we increases the number of groups from 1 to 32 while fixing the total number of users in a cell to 32. T_i is the throughput of user i. MAX is optimized for this metric.

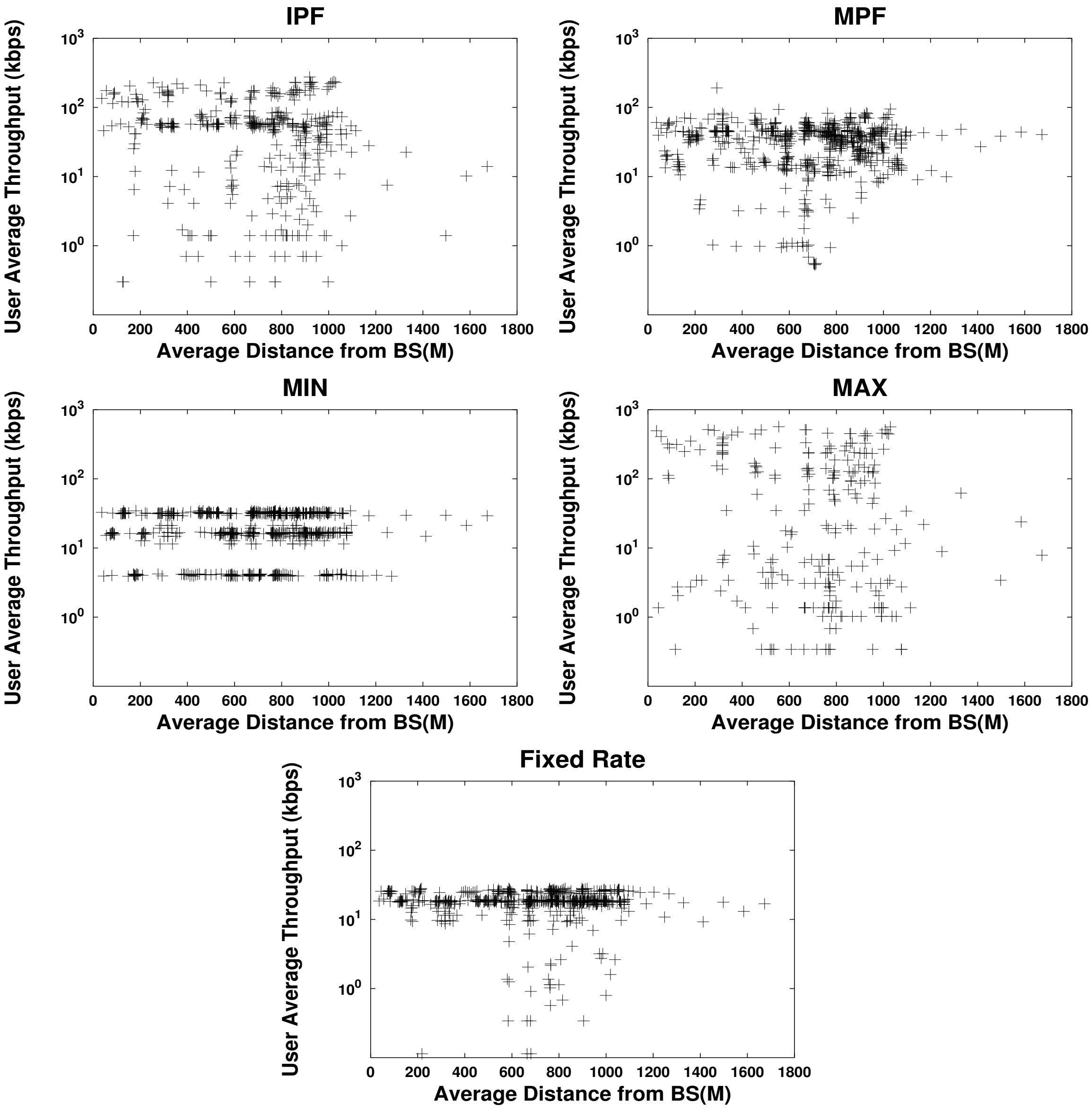

Fig. 5. The distribution of user throughput taken at each three-second interval for various schedulers. Users are sorted over X-axis by their distance to the base station. The key point in these figures is that the distribution of user throughput tends to follow the characteristics of tradeoff that each scheduler makes regarding the throughput and fairness of users.

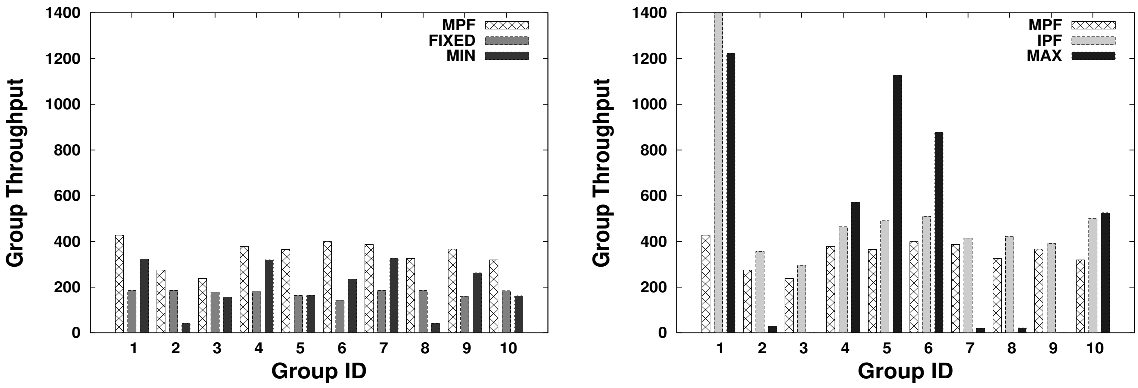

Fig. 6. The distribution of group throughput at the end of simulation for various schedulers. The key point in these figures is that the distribution of group throughput tends to follow the characteristics of tradeoff that each scheduler makes regarding the throughput and fairness of groups.