A Self-Optimization Method for Coverage-Capacity Optimization in OFDMA networks with MIMO

Richard Combes*
France Telecom Research and Development
38/40 rue du General Leclerc
92794 Issy-les-Moulineaux
richard.combes@orange-ftgroup.com

Zwi Altman
France Telecom Research and Development
38/40 rue du General Leclerc
92794 Issy-les-Moulineaux
zwi.altman@orange-ftgroup.com

Eitan Altman
INRIA Sophia Antipolis
06902 Sophia Antipolis, France
Eitan.Altman@sophia.inria.fr

ABSTRACT

Self-organizing networks (SON) is currently seen as a key lever to improve network performance and simplify its management. This paper considers α-fair schedulers in an Orthogonal Frequency-Division Multiple Access (OFDMA) network. The convergence of the α-fair scheduler is analyzed. Closed-form formulas are given for certain cases to calculate the scheduling gain, as well as a Monte-Carlo method, for a MIMO channel. A capacity-coverage algorithm based on the α-fair schedulers using observable Key Performance Indicators (KPIs) is proposed. The algorithm is implemented in a large scale network simulator. It is shown that notable coverage gains are achieved at the expense of very small capacity losses.

Keywords

Self-Optimizing Networks, OFDMA, MIMO, scheduling gain, α-fair.

1. INTRODUCTION

SON receives increasing importance in Next Generation (NG) Radio Access Networks (RAN) such as Long Term Evolution (LTE), LTE Advanced and WiMax 802.16m,([2], [1]). SON mechanisms including self-configuration, self-optimization and self-healing will allow to simplify network management, reduce its cost of operation, and improve its performance. A detailed description of the requirements for SON mechanisms can be found in [11].

Self-optimization aims at adapting the network to variations in traffic, in propagation conditions and other operation conditions such the introduction of a new Base Station (BS). It is used to dynamically adapt radio resource management functionalities such as Inter-Cell Interference Coordination (ICIC), mobility management, and more recently, energy saving [3]. Self-optimization processes can enhance the perceived Quality of Service (QoS) and network performance, and provide a lever to enforce operator business strategies.

On-line self-optimization algorithms have to meet strict requirements in terms of processing speed, and must be implemented in the control plane in a distributed and scalable fashion in order to be in line with the distributed architecture of future RANs. Their stability is also a crucial question. Off-line self-optimization algorithms operate on slower time scales with less constraints: they can be centralized and use data from both management and control planes.

One of the challenging problems in SON is coverage-capacity optimization, i.e. designing self-optimizing algorithms that achieve optimal trade-offs between coverage and capacity. It can be seen as a form of fairness: we consider a service in which users have a minimum bitrate requirement to ensure good QoS, and we try to satisfy the maximum number of users while minimizing the corresponding capacity losses.

Different mechanisms can be considered to dynamically improve coverage and capacity, such as ICIC ([15], [6]), scheduling [5], and the combination of such mechanisms. In a previous work, we have shown how the scheduling strategy can be dynamically adapted to optimize coverage and capacity in the downlink of a Time Division Multiple Access (TDMA) network [5]. The α-fairness framework has been considered for the self-optimization algorithm that optimizes the number of covered users while minimizing the associated capacity losses. To evaluate the performance of the scheduler for a specific α-fair parameter in a network simulator, one needs to compute the scheduling gain.

The objective of this paper is to provide new results on α-fair scheduling in the context of OFDMA systems and use the resulting model to devise a self-optimization algorithm based on the α-fair scheduler. The paper contributions are the following:

- the scheduling rule for an α-fair OFDMA scheduler is

*This work has been partially supported by the Agence Nationale de la Recherche within the project ANR-09-VERS0:

VALUETOOLS 2011, May 16-20, Paris, France

DOI 10.4108/icst.valuetools.2011.245615

stated;

- the calculation of the scheduling gain is introduced for a MIMO channel;
- the optimality of the scheduling scheme is demonstrated;
- the self-optimizing algorithm for coverage-capacity optimization is presented and is tested on a large scale network simulator.

The paper is organized as follows: Section 2 presents the α-fair scheduler for an OFDMA system, including a heuristic justification for the scheduling rule. Section 3 describes the fast fading Multiple Input Multiple Output (MIMO) channel model using the Vertical Bell Labs Space-Time (V-BLAST) architecture, and an approximation of the asymptotic capacity distribution using the random matrix theory. The main steps for proving the optimality of the α-fair allocation rule for an OFDMA system are summarized in Section 4. Section 5 describes the methodology for calculating the scheduling gain for the α-fair scheduler. Section 6 proposes a coverage-capacity self-optimization algorithm, followed by simulation results in Section 7. Section 8 concludes the paper.

2. ALPHA-FAIR SCHEDULING

2.1 Definitions and notations

We consider a cell of an OFDMA network such as LTE or WiMAX with N users, and the total available bandwidth W is divided in K Physical Resource Blocks (PRBs). We have a set of scheduling instants $(t_m)_{m\in\mathbb{N}}$, and at each instant t_m, a scheduler chooses a user for transmission on each PRB. We define the scheduling policy P, with $P_{t_m}^{(k)} = i$ if user i is selected at time t_m to transmit on PRB k. We define $r_{i,t_m}^{(k)}$ the instantaneous throughput of user i at time t_m on PRB k, and $\overline{r}_{i,t_m}^{(k)}$ the mean throughput allocated to user i during the time interval $[t_0, t_m]$ on PRB k. We denote the Signal to Interference plus Noise Ratio (SINR) of user i on PRB k by $S_i^{(k)}$. Let $\epsilon > 0$ denote a small averaging parameter and define $\overline{r}_{i,t_m}^{(k)}$ by the following recursive equation:

$$\overline{r}_{i,t_{m+1}}^{(k)} = (1-\epsilon)\overline{r}_{i,t_m}^{(k)} + \epsilon\delta_{P_{t_{m+1}}^{(k)},i} r_{i,t_{m+1}}^{(k)} \tag{1}$$

δ being Kronecker's delta.

This definition for the mean allocated throughput is better than the one using an arithmetic mean (by replacing ϵ in (1) by $\frac{1}{m}$) because it induces a "decay" of past observed values. If we assume that $\overline{r}_{i,t_0}^{(k)} = 0\ \forall(i,k)$, equation (1) can also be written:

$$\overline{r}_{i,t_m}^{(k)} = \epsilon\sum_{j=0}^{m}(1-\epsilon)^{m-j}\delta_{P_{t_j}^{(k)},i} r_{i,t_j}^{(k)} \tag{2}$$

Finally, we define $\overline{r}_{i,t_m}$-the mean throughput allocated to user i during the time interval $[t_0, t_m]$ by:

$$\overline{r}_{i,t_m} = \sum_{k=1}^{K}\overline{r}_{i,t_m}^{(k)} \tag{3}$$

2.2 α-fair scheduler

We define the α-fair scheduler as in [10]. Let M be a given number of scheduling periods. The α-fair scheduler is the allocation strategy that maximizes the following utility function (with $d > 0$ a small parameter to avoid singularity at 0):

$$U = \begin{cases} \sum_{i=1}^{N}\log(d+\overline{r}_{i,t_M}) & , \ \alpha = 1 \\ \sum_{i=1}^{N}\frac{(\overline{r}_{i,t_M}+d)^{1-\alpha}-1}{1-\alpha} & , \ \alpha \neq 1 \end{cases} \tag{4}$$

2.3 Scheduling rule

The maximization problem (4) is a priori non trivial, so we first give a heuristic justification for the scheduling strategy. The rigorous proof of the optimality of this rule will be given in Section 4. Choosing user i for transmitting at time t_{m+1} on PRB k results in the following increase of utility:

For $\alpha = 1$:

$$\log\left((1-\epsilon)\overline{r}_{i,t_m} + \epsilon r_{i,t_{m+1}}^{(k)} + d\right) - \log(\overline{r}_{i,t_m}+d) = \epsilon\frac{r_{i,t_{m+1}}^{(k)} - \overline{r}_{i,t_m}}{\overline{r}_{i,t_m}+d} + o(\epsilon) \tag{5}$$

The utility decrease for the other users is:

$$\log\left((1-\epsilon)\overline{r}_{i,t_m}\right) - \log(\overline{r}_{i,t_m}) = -\epsilon\frac{\overline{r}_{i,t_m}}{\overline{r}_{i,t_m}+d} + o(\epsilon) \tag{6}$$

We add (5) and (6):

$$(\Delta U)_i = \epsilon\left(\frac{r_{i,t_{m+1}}^{(k)}}{\overline{r}_{i,t_m}+d} - \sum_{l=1}^{N}\frac{\overline{r}_{l,t_m}}{\overline{r}_{l,t_m}+d}\right) + o(\epsilon) \tag{7}$$

If $\alpha \neq 1$, the utility increases by:

$$\frac{1}{1-\alpha}\left[\left((1-\epsilon)\overline{r}_{i,t_m} + \epsilon r_{i,t_{m+1}}^{(k)} + d\right)^{1-\alpha} - (\overline{r}_{i,t_m}+d)^{1-\alpha}\right] = \epsilon\frac{r_{i,t_{m+1}}^{(k)} - \overline{r}_{i,t_m}}{(\overline{r}_{i,t_m}+d)^{\alpha}} + o(\epsilon) \tag{8}$$

And the other utilities decrease by:

$$\frac{1}{1-\alpha}\left[((1-\epsilon)\overline{r}_{i,t_m}+d)^{1-\alpha} - (\overline{r}_{i,t_m}+d)^{1-\alpha}\right] = -\epsilon\frac{\overline{r}_{i,t_m}}{(\overline{r}_{i,t_m}+d)^{\alpha}} + o(\epsilon) \tag{9}$$

We add (8) and (9):

$$(\Delta U)_i = \epsilon\left[\frac{r_{i,t_{m+1}}^{(k)}}{(\overline{r}_{i,t_m}+d)^{\alpha}} - \sum_{l=1}^{N}\frac{\overline{r}_{l,t_m}}{(\overline{r}_{l,t_m}+d)^{\alpha}}\right] + o(\epsilon) \tag{10}$$

Therefore, for ϵ small enough, the optimal user to schedule for transmission at time t_{m+1} on PRB k is:

$$i^{*(k)} = \arg\max_{0\le i\le N}\frac{r_{i,t_{m+1}}^{(k)}}{(\overline{r}_{i,t_m}+d)^{\alpha}} \tag{11}$$

In the case of $\alpha = 1$, the rule becomes the well-known Proportional Fair (PF) scheduler: we choose the user with the best instantaneous throughput to average throughput ratio.

3. MIMO CHANNEL

3.1 MIMO capacity

MIMO systems have received much attention since the seminal work of Telatar ([13]), and future LTE networks are expected to feature at least MIMO 2×2 (2 receive antennas and 2 transmit antennas). We use the following notations: let n_t denote the number of transmit antennas, n_r - the number of receive antennas, I_{n_r} - the $n_r \times n_r$ identity matrix, H - the $n_r \times n_t$ channel matrix. We assume that all entries of H are standard complex normal random variables and are all independent. We consider the V-BLAST architecture in which the transmitter does not know the instantaneous channel realization, but knows its distribution. With the model we are considering for H, the instantaneous capacity of user i on PRB k for a given channel realization H is then (see [14](p337)):

$$C_i^{(k)} = \log_2 \left[\det \left(I_{n_r} + \frac{S_i^{(k)}}{n_t} H H^* \right) \right] \tag{12}$$

The ergodic capacity is then $\mathbb{E}[C_i^{(k)}]$.

3.2 Asymptotic distribution of the capacity

It has been shown in [8] that with the previous assumptions the capacity is asymptotically normal when $n_{min} = \min(n_t, n_r) \to +\infty$, with the following mean and variance:

$$(C_i^{(k)} - n_t \mu_{C_i^{(k)}}) \to \mathcal{N}(0, \sigma^2_{C_i^{(k)}}) \tag{13}$$

$$\beta = \frac{n_r}{n_t}$$

$$\alpha = \frac{1}{2} \left[1 + \beta + \frac{1}{S_i^{(k)}} - \sqrt{\left(1 + \beta + \frac{1}{S_i^{(k)}}\right)^2 - 4\beta} \right]$$

$$\mu_{C_i^{(k)}} = \frac{1}{\log(2)} \left[\beta \log(1 + S_i^{(k)} - S_i^{(k)} \alpha) + \log(1 + S_i^{(k)} \beta - S_i^{(k)} \alpha) - \alpha \right]$$

$$\sigma^2_{C_i^{(k)}} = -\frac{\log\left(1 - \frac{\alpha^2}{\beta}\right)}{\log(2)^2}$$

We now compare the distribution of the capacity of a MIMO channel with its asymptotic distribution given by (13), for $n_t = n_r = 2$. We draw the H matrix 10000 times and calculate the corresponding capacity distribution with formula (12), which we compare to the Gaussian distribution with mean and variance given by (13). Figure 1 shows the comparison of the mean of the two distributions for different values of the SINR. Figure 2 shows the comparison of the cumulative distribution function (c.d.f) of the two distributions for a SINR of 5dB. We can see on those two figures that the values obtained by the Gaussian approximation are very close to the simulated values obtained by drawing H matrices. Hence approximating the distribution of the capacity by a Gaussian distribution is reasonable, even when $n_t = n_r = 2$.

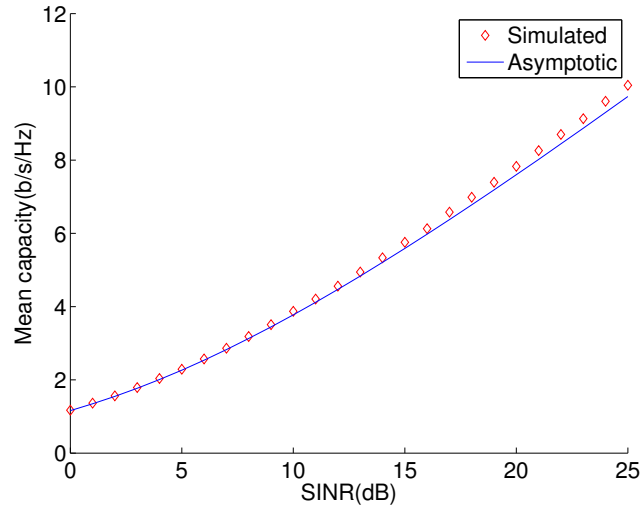

Figure 1: Mean capacity for a MIMO 2x2, comparison between asymptotic distribution (13) and simulations.

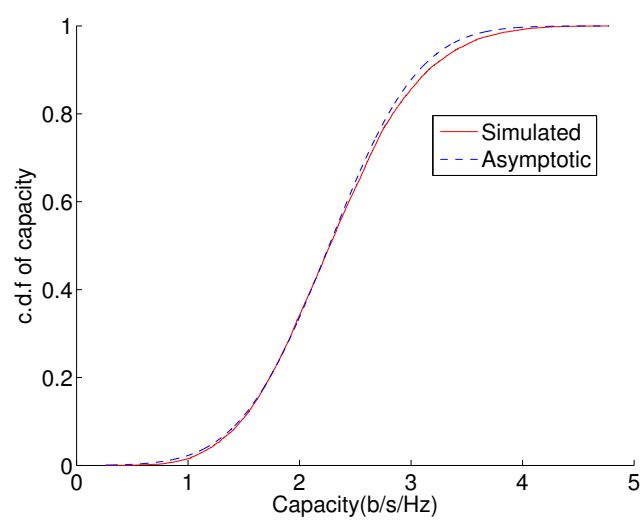

Figure 2: c.d.f for a MIMO 2x2, comparison between asymptotic distribution (13) and simulations, for SINR 5dB.

3.3 MIMO and OFDMA

We assume that $r_{i,t_m}^{(k)}$ is distributed like $C_i^{(k)}$, and that $r_{i,t_m}^{(k)}$ is independent of $r_{j,t_m}^{(k)}$ $\forall k$, $\forall t_m$ and $r_{i,t_m}^{(k_1)}$ is independent of $r_{i,t_m}^{(k_2)}$ $\forall i$, $\forall t_m$.

The independence in time is a valid assumption if the time interval between two scheduling instants $t_{m+1} - t_m$ is large enough as stated in [7]. Namely, the autocorrelation of the channel fading between t and $t+\tau$ is $J_0(\omega_M \tau)$, where J_0 is the 0-th order Bessel function and ω_M the maximum Doppler shift, and $|J_0(x)| \underset{x\to+\infty}{\to} 0$. Independence between PRBs holds as long as the Doppler shift $\frac{v}{\lambda}$ is very small compared with the size of a PRB, with λ the wavelength and v the speed of the mobile. For a frequency of $1GHz$, and $v = 10km/h$, $\frac{v}{\lambda} = 9Hz$, which is way smaller than the size of a PRB.

4. CONVERGENCE ANALYSIS

4.1 Stochastic Approximation

In this section we give a convergence analysis of the α-fair scheduler, based on the demonstration given in [5] for the TDMA case. We begin by stating stochastic approximation results that link the α-fair scheduler with the asymptotic behavior of a particular Ordinary Differential Equation (ODE). We consider n users, and use the following conventions: if $(x,y) \in \mathbb{R}^n \times \mathbb{R}^n$, we say that $x \leq y$ if $x_i \leq y_i$, $1 \leq i \leq n$. We denote by xy the component-wise product of x and y i.e $(xy)_i = x_i y_i$, $1 \leq i \leq n$.

The following notations are used: let (a,b) two vectors of $\mathbb{R}^n$, $Q = \{x \in \mathbb{R}^n | a \leq x \leq b\}$ and $\Pi_Q[x] = \underset{y\in Q}{\text{argmin}} \, ||x-y||$ the projection on Q with respect to the Euclidean norm. Let $\theta \in \mathbb{R}^n$, $(\epsilon_{t_m})_{m\in\mathbb{N}}$ be a sequence of step-sizes, $(Y_{t_m}(\theta))_{m\in\mathbb{N}}$, a sequence of random variables in $\mathbb{R}^n$ and g a function defined by $\mathbb{E}[Y_{t_m}(\theta)] = g(\theta)$. We then define the sequence $(\theta_{t_m})_{m\in\mathbb{N}}$ by the following equation:

$$\theta_{t_{m+1}} = \Pi_Q[\theta_{t_m} + \epsilon_{t_m} Y_{t_m}(\theta_{t_m})] \tag{14}$$

We assume the following: $(Y_{t_m}(\theta))_{m\in\mathbb{N}}$ are independent and identically distributed (i.i.d), $\sup_\theta \mathbb{E}[Y_{t_m}(\theta)^2] < +\infty$ and all solutions to the ODE $\dot{\theta} = g(\theta)$ converge to θ^* in the interior of Q, for all initial conditions. If those assumptions are verified we have that ([9] (Theorem 2.1, page 127) and [4] (Theorem 3, Page 106)):

Theorem 1. *If $\epsilon_{t_m} > 0$, $\sum_{m\in\mathbb{N}} \epsilon_{t_m} = +\infty$ and $\sum_{m\in\mathbb{N}} \epsilon_{t_m}^2 < +\infty$ then $\theta_{t_m} \underset{m\to+\infty}{\to} \theta^*$ almost surely.*

Theorem 2. *If $\epsilon_{t_m} = \epsilon > 0$, then there exists a constant $K_1 > 0$ such that $\limsup_{m\to+\infty} \mathbb{E}[||\theta_{t_m} - \theta^*||^2]^{\frac{1}{2}} \leq K_1\sqrt{\epsilon}$*

4.2 α-fair scheduling

We can see that the α-fair scheduler is a particular case of (14), with $Q = (\mathbb{R}^+)^n$, $\epsilon_{t_m} = \epsilon > 0$, $(\theta_{t_m})_i = \overline{r}_{i,t_m}$, $1 \leq i \leq n$, and $Y_{t_m} = \sum_{k=1}^K r^{(k)} I_{\arg\max(\frac{r^{(k)}}{(d+\theta)^\alpha})}$ where $r^{(k)}$ is a random vector distributed like the throughput of all users at time t_m on PRB k and (I_i) the vector in $\mathbb{R}^n$ whose components are all equal to 0 except the i-th which is equal to 1. The ODE is then:

$$\dot{\theta} = h(\theta) - \theta \tag{15}$$

Where h is given by:

$$h(\theta) = \sum_{k=1}^K h^{(k)}(\theta) = \sum_{k=1}^K \mathbb{E}[r^{(k)} I_{\arg\max(\frac{r^{(k)}}{(d+\theta)^\alpha})}] \tag{16}$$

4.3 Convergence of the ODE

We have proved in [5] that $h^{(k)}(\theta)$ is globally Lipschitz continuous, therefore $h(\theta)$ is globally Lipschitz continuous as well. Since $x \leq y$ and $x_i = y_i$ implies $h^{(k)}(x)_i \leq h^{(k)}(y)_i$, it also implies $h(x)_i \leq h(y)_i$. Therefore the exact same demonstration given in [5] proves that all solutions of the ODE converge to the same limit θ^* as described by Theorem 2.

4.4 Optimality

Now let us prove that θ^* is the unique maximizer of U in the set of achievable mean throughputs. Differentiating $U(\theta(t))$ with respect to t yields:

$$\dot{U}\,(\theta(t)) = \sum_{i=1}^n \frac{\sum_{k=1}^K h_i^{(k)}(\theta(t)) - \theta_i(t)}{(d+\theta_i(t))^\alpha} \tag{17}$$

Therefore by the same argument as [5] we have that θ^* is the global optimum of U in the set of achievable mean throughputs.

5. SCHEDULING GAIN

According to the results in Section 4, we now know that the mean throughput of a user can be calculated by evaluating a certain integral that depends on the SINR for each PRB and the distribution chosen for the capacity. We will now show how to calculate the scheduling gain in a OFDMA system based on a per PRB approach.

5.1 Link with TDMA scheduling

It is noted that the channel is not frequency selective on a long-time scale, and that whenever a BS transmits on a PRB, it transmits at full power, namely we consider a Reuse 1 scheme. Therefore the mean SINR of a user is the same on all PRBs, $S_i^{(k_1)} = S_i^{(k_2)}$, $\forall i$, $k_1 \neq k_2$. In order to calculate the mean throughput of the α-fair scheduler, we now introduce a fictive scheduler, the Per Physical Resource Block Scheduler (PPRBS), which chooses for transmission on PRB k at time t_{m+1} the user that maximizes:

$$i^* = \arg\max_{0\leq i\leq N} \frac{r_{i,t_{m+1}}^{(k)}}{(\overline{r}_{i,t_m}^{(k)} + d)^\alpha} \tag{18}$$

We can see that (18) is significantly different from (11) because it uses the past allocated throughput by PRB instead of the sum of the allocated throughputs for all PRBs. In other words the PPRBS ignores the resources allocated on other PRBs, and behaves exactly as if we were applying K TDMA schedulers in parallel, one for each PRB.

We have proved the convergence of the α-fair scheduler so we can consider the limit of the average throughput it allocates on PRB k $\overline{r}_{i,+\infty}^{(k)}$. By symmetry, we also have that:

$\overline{r}_{i,+\infty}^{(k')} = \overline{r}_{i,+\infty}^{(k)}$, $\forall k, k', i$. Therefore $\overline{r}_{i,+\infty} = K\overline{r}_{i,+\infty}^{(k)}$, and the scheduling rule (11) becomes:

$$i^* = \arg \max_{0 \le i \le N} \frac{r_{i,t_{m+1}}^{(k)}}{(K\overline{r}_{i,t_m}^{(k)} + d)^\alpha} \tag{19}$$

Therefore, for d sufficiently small, the α-fair scheduler and the PPRBS behave the same way and we can calculate the throughput of the α-fair scheduler by summing over all PRBs the mean throughput allocated by the PPRBS.

5.2 Scheduling gain

We can now calculate the mean throughput allocated by the α-fair scheduler: equation (13) states that $r_{i,t_{m+1}}^{(k)} \sim \mathcal{N}(W_{PRB} n_t \mu_{C_i^{(k)}}, W_{PRB}^2 \sigma_{C_i^{(k)}}^2)$, where n_t is the number of antennas and W_{PRB} the bandwidth corresponding to one PRB. It is noted that there is a non-null probability that $r_{i,t_{m+1}}^{(k)} < 0$, which does not make sense physically, and is due to the fact that assuming (13) for n_t finite is only an approximation, and $\mathbb{P}\left[r_{i,t_{m+1}}^{(k)} \le 0\right]$ vanishes when n_{min} goes to infinity.

Let $\overline{r}_{i,+\infty,\alpha}$ denote the average throughput allocated to user i by the α-fair scheduler, and $f_i^{(k)}$ the probability density function (p.d.f) of $r_{i,t_m}^{(k)}$. $\overline{r}_{i,+\infty,\alpha}$ obeys the following integral equation:

$$\overline{r}_{i,+\infty,\alpha} = \sum_{k=1}^{K} \int_{-\infty}^{+\infty} z\mathbb{P}\left[\frac{z}{\overline{r}_{i,+\infty,\alpha}^{\alpha}} \ge \max_{j \ne i}\left(\frac{r_{j,t_m}^{(k)}}{\overline{r}_{j,+\infty,\alpha}^{\alpha}}\right)\right] f_i^{(k)}(z)dz \tag{20}$$

We also define G_α the scheduling gain of the α-fair scheduler by:

$$G_\alpha = \frac{\overline{r}_{i,+\infty,\alpha}}{\overline{r}_{i,+\infty,RR}} \tag{21}$$

where $\overline{r}_{i,+\infty,RR}$ denotes the mean throughput allocated to user i by a Round Robin (RR) scheduler. We solve (20) with the following method: we first solve it for $K = 1$, then we use the argument exposed previously to say that the α-fair scheduler and the PPRBS behave the same way, and obtain the throughput by summing over all PRBs.

5.2.1 Notation

In order to reduce the notational burden, we adopt the following conventions to refer to the quantities defined in (13):

$$\mu_i = W_{PRB} n_t \mu_C(S_i) \tag{22}$$

$$\sigma_i = W_{PRB} \sigma_C(S_i) \tag{23}$$

Where S_i is the mean SINR of user i on a PRB, since we have assumed that the mean SINR is the same on all PRBs.

5.2.2 PF

We consider the PF scheduler, that is $\alpha = 1$, and first assume that there is only 1 PRB. Since $r_{i,t_m}^{(k)} \sim \mathcal{N}(\mu_i, \sigma_i^2)$, the scheduler picks the user that maximizes $\frac{r_{i,t_m}^{(1)}}{\mu_i}$. Let $F(z) = \mathbb{P}[Z \le z]$, $Z \sim \mathcal{N}(0,1)$. The probability that user i is chosen can be written:

$$\mathbb{P}\left[\frac{z}{\overline{r}_{i,+\infty,1}} \ge \max_{j \ne i}\left(\frac{r_{j,t_m}^{(k)}}{\overline{r}_{j,+\infty,1}}\right)\right] = \prod_{j \ne i} F\left(z\frac{\mu_i \sigma_j}{\mu_j \sigma_i}\right) \tag{24}$$

Replacing in (20) we get $(\overline{r}_{i,+\infty,1})_{1\,PRB}$, the throughput on one PRB:

$$(\overline{r}_{i,+\infty,1})_{1\,PRB} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} (z\sigma_i + \mu_i) \left[\prod_{j \ne i} F\left(z\frac{\mu_i \sigma_j}{\mu_j \sigma_i}\right)\right] e^{-\frac{z^2}{2}} dz \tag{25}$$

Now for the case $K \ne 1$, we can use the previous argument that the PPRBS and the α-fair scheduler behave the same way, and summing over all PRBs yields the result:

$$\overline{r}_{i,+\infty,1} = \frac{K}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} (z\sigma_i + \mu_i) \left[\prod_{j \ne i} F\left(z\frac{\mu_i \sigma_j}{\mu_j \sigma_i}\right)\right] e^{-\frac{z^2}{2}} dz \tag{26}$$

5.2.3 Max Throughput (MTP)

We now consider the MTP scheduler ($\alpha = 0$), and we first assume $K = 1$. The scheduler picks the user with the best instantaneous throughput, therefore the probability to choose user i is:

$$\mathbb{P}\left[z \ge \max_{j \ne i}\left(r_{j,t_m}^{(1)}\right)\right] = \prod_{j \ne i} F\left(\frac{\mu_i - \mu_j + z\sigma_i}{\sigma_j}\right) \tag{27}$$

The throughput for one PRB is then:

$$\overline{r}_{i,+\infty,0} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} (z\sigma_i + \mu_i) \left[\prod_{j \ne i} F\left(\frac{\mu_i - \mu_j + z\sigma_i}{\sigma_j}\right)\right] e^{-\frac{z^2}{2}} dz \tag{28}$$

As previously, we sum over all PRBs to obtain the result:

$$\overline{r}_{i,+\infty,0} = \frac{K}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} (z\sigma_i + \mu_i) \left[\prod_{j \ne i} F\left(\frac{\mu_i - \mu_j + z\sigma_i}{\sigma_j}\right)\right] e^{-\frac{z^2}{2}} dz \tag{29}$$

5.2.4 Max-Min Fair (MMF)

The last analytically tractable case is the MMF scheduler ($\alpha = +\infty$). As done previously we start by $K = 1$ and the scheduling rule becomes:

$$i^* = \arg \min_{0 \le i \le N} \overline{r}_{i,t_m} \tag{30}$$

First, suppose that there exists i and j such that $\overline{r}_{i,+\infty,+\infty} > \overline{r}_{j,+\infty,+\infty}$. This means that after a certain time, user i will never be scheduled for transmission, hence $\overline{r}_{i,+\infty,+\infty} = 0$ and $\overline{r}_{j,+\infty,+\infty} = 0$, which contradicts our first hypothesis. Hence, we have proved that the MMF scheduler gives the same mean throughput to all users. We also notice that the scheduling decision does not depend on the instantaneous

throughput, and so there exists some weights $(p_i)_{1\leq i\leq N}$ so that:

$$\overline{r}_{i,+\infty,+\infty} = p_i\mu_i \tag{31}$$

$$\sum_{i=1}^{N} p_i = 1 \tag{32}$$

$$p_i\mu_i = p_j\mu_j \ \forall i,j \tag{33}$$

Therefore $p_i = \frac{1}{\mu_i}\frac{1}{\sum_{j=1}^{N}\frac{1}{\mu_j}}$, and:

$$\overline{r}_{i,+\infty,+\infty} = \frac{1}{\sum_{j=1}^{N}\frac{1}{\mu_j}} \tag{34}$$

We then sum over all PRBs:

$$\overline{r}_{i,+\infty,+\infty} = \frac{K}{\sum_{i=1}^{N}\frac{1}{\mu_i}} \tag{35}$$

5.2.5 Monte-Carlo Method

For a general α there is no analytical formula, and we provide the numerical method described in Table 1. It is noted

1. $\overline{r}_{i,t_0,\alpha} = 0 \ \forall i$

For t_m **from** t_0 **to** T**:**

2. Draw the channel for each user on each PRB $(r^{(k)}_{i,t_m})_{0\leq i\leq N, 0\leq k\leq K}$

3. $i^{(k)} = \arg\max_{0\leq i\leq N} \frac{r^{(k)}_{i,t_m}}{\overline{r}^{\alpha}_{i,t_m,\alpha}}$

For i **from** 1 **to** N**:**

4. $\overline{r}_{i,t_{m+1},\alpha} = (1-\epsilon_n)\overline{r}_{i,t_m,\alpha}$
$+\epsilon_n \sum_{k=1}^{K} r^{(k)}_{i,t_m}\mathbf{1}_{i=i^{(k)}}$

End For

End For

Table 1: Numerical method for calculating $\overline{r}_{i,+\infty,\alpha}$

that that step 2 might take two forms: it is either possible to draw the MIMO channel matrix for each user and each PRB, or draw a Gaussian random variable with mean and variance given by (13), which makes the computation considerably faster, if the number of antennas is large. It is noted that all those random variables are independent according to our model. Furthermore, choosing $\epsilon_n = \epsilon > 0$ a small constant or $\epsilon_n = \frac{1}{n}$ both guarantee convergence to the α-fair allocation.

5.3 Simulation results

We now compare the formulas stated above with the actual mean throughput obtained by simulating the MIMO channel and the α-fair scheduler. We choose $K = 12$, $W_{PRB} = 180kHz$ and a scheduling interval of $1ms$, as defined by the LTE standard. We simulate 1000 scheduling intervals, and a 95% confidence interval is provided, assuming normality of the estimates.

Figure 3 shows the throughput per user of a PF scheduler with $S_i^{(k)} = 5dB, \forall i,k$. Figure 4 shows the mean throughput of the MTP scheduler with 2 users, when one of the two users

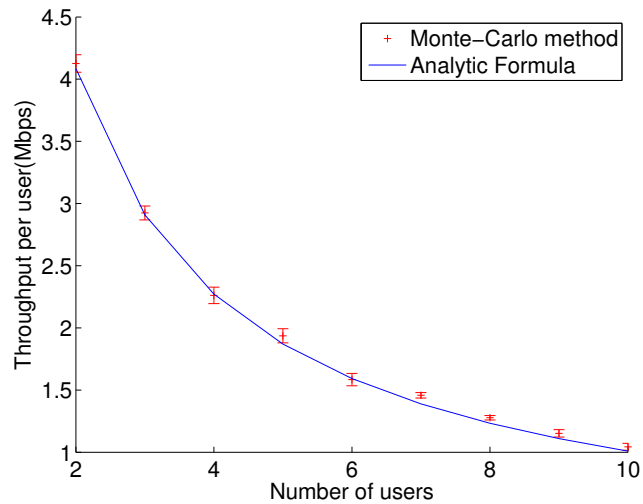

Figure 3: PF scheduler, mean throughput per user, with $S_i^{(k)} = 5dB$, $\forall i,k$. Comparison between simulations and formula (26).

has a better SINR. We can see that the user with poorer conditions has a significantly smaller mean throughput, and gets almost nothing when $S_1^{(k)} = 5dB$ and $S_2^{(k)} = 12dB$ $\forall k$. Figure 5 shows the mean throughput of user 1 with a MMF scheduler when $S_{2i-1}^{(k)} = 5dB$ and $S_{2i}^{(k)} = 12dB$, $\forall i,k$. We can see that when a user with good conditions enters the system, the max-min mean throughput is less affected than when a user with poor channel conditions enters, since those users need to be scheduled much more often to give the same mean throughput to all users. This idea is also relevant when we consider which users to admit in a network if we are willing to deliver some minimal QoS to all users. It is also noted that the analytic formulas are very accurate when compared to simulations.

6. COVERAGE CAPACITY SELF-OPTIMIZATION

6.1 Algorithm

Based on the scheduling gain calculation of the previous section, we propose a simple and efficient SON algorithm that optimizes cell-coverage while minimizing capacity losses by adjusting α dynamically. This algorithm is the adaptation to OFDMA of the one introduced in [5]. We say that a user is covered if his mean throughput is higher than a certain fixed threshold Th_{min}, which is a parameter of the service we are considering, for example the minimal throughput to watch a video with the lowest quality.

First let us state the optimization objective: we consider a particular service with the corresponding Th_{min} and we want to change the α parameter dynamically in order to cover the maximum number of users, using the above definition for coverage. However, we have to be careful since increasing α can potentially increase the number of covered users, but also diminishes the global cell throughput. Therefore we want to find the minimal α that covers the maximum number of users.

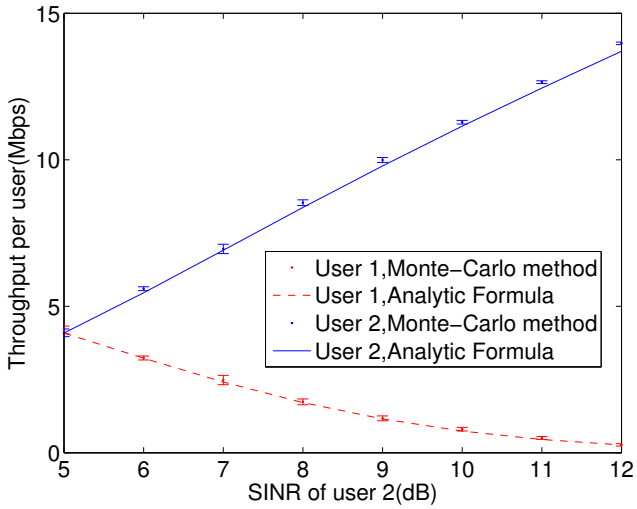

Figure 4: MTP scheduler, 2 users, $S_1^{(k)} = 5dB$, $\forall k$. Comparison between simulations and formula (29).

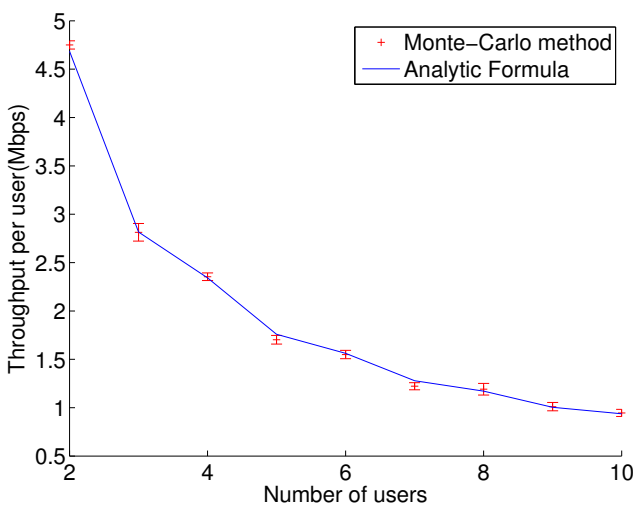

Figure 5: MMF scheduler, mean throughput of user 1 with $S_{2i-1}^{(k)} = 5dB$ and $S_{2i}^{(k)} = 12dB$, $\forall i, k$.

To this end, the formula for the scheduling gain with $\alpha = +\infty$ is of particular interest: if $\alpha = +\infty$ results in covering all users, this means that we can cover everyone providing that α is large enough. If nobody is covered, namely the users with bad SINR will never be covered, and we should not allocate any resource to them.

In order to determine the users that can be covered with large enough α, we ignore the user with the worst SINR, recalculate the $\alpha = +\infty$ throughput and keep doing so until we are able to cover everyone.

The algorithm proceeds the following way: at each iteration it observes the number of covered users, then it determines the users that can be covered using the technique stated above, and finally the α is adjusted. If some of the users that could have been covered were not covered, the α is increased, and if all coverable users have been covered, the α is diminished with a small probability P_ϵ, and stays the same with probability $1 - P_\epsilon$. The idea is that the environment might have changed, and that the current α might not be the lowest that enables us to cover all coverable users. P_ϵ therefore shall be chosen to reflect the speed at which the environment changes.

The following notations are used: we consider BS s; α_s is the value of α for s, N_s - the number of users that s can cover and $\tilde{N}_s$ - the number of users effectively covered at the last period. $(\alpha^{(j)})_{1 \le j \le J_{max}}$ is the allowed set of values of α, e.g. $\{1, ..., 5\}$ in the present work. j_s is the index of the current α, namely $\alpha_s = \alpha^{(j_s)}$. The algorithm is described in Table 2.

For each BS s:
Initial phase:
1. Calculate N_s using (Table 3)
2. Try every $\alpha_s \in (\alpha^{(j)})_{1 \le j \le J_{max}}$ once
3. Choose the minimal j_s so that $\alpha_s = \alpha^{(j_s)}$ that covers N_s users.

Repeat:
4. Calculate N_s using (Table 3)
5. Set $\alpha_s = \alpha^{(j_s)}$ and observe resulting $\tilde{N}_s$

If $\tilde{N}_s < N_s$:
6. $j_s \leftarrow \min(j_s + 1, J_{max})$

If $n_k = N_k$:
7. $j_s \leftarrow \begin{cases} \max(j_s - 1, 1) \text{ with probability } P_\epsilon \\ j_s \text{ with probability } (1 - P_\epsilon) \end{cases}$

Table 2: Capacity coverage algorithm

It is noted that in Table 3 it is sufficient to calculate the throughput of a user in $i \in \{1, ..., N\} \setminus I$ since the MMF scheduler allocates the same throughput to all users in $i \in \{1, ..., N\} \setminus I$ and allocates 0 to users in I.

It is noted that this algorithm has all the necessary features to be a robust and implementable SON algorithm: it is decentralized since each station adjusts its own parameters according to its own KPIs without any communication with neighboring cells; it is not computationally demanding; and it is scalable since the introduction of new base stations

Initial phase:
1. $I = \emptyset$
2. Calculate $\overline{r}_{i,\infty,\infty}$ for a certain $i \in \{1, ..., N\} \setminus I$ using (35)
While $\overline{r}_{i,\infty,\infty} < Th_{min}$**:**
3. $i^* = \arg\min_{i \in \{1,...,N\} \setminus I} S_i$
4. Add i^* to I
5. Calculate $\overline{r}_{i,\infty,\infty}$ for a certain $i \in \{1, ..., N\} \setminus I$ ignoring users in I, using (35)
Result:
6.$N_k = N - \|I\|$

Table 3: Calculation of N_k

does not disturb its functioning.

6.2 Admission Control

It shall be noted that formula (35) is also useful to define an admission control rule. Given the SINR of the users in a cell, if a new user arrives, we can calculate the throughput of the MMF scheduler with (35) and determine whether we are able to cover this user with α sufficiently large. If it is not the case the new user shall not be admitted. The benefit of such an admission rule over traditional methods is that we can be sure that we will always be able to cover all users if they do not move too fast, so that their SINR does not change too drastically over time. Furthermore since (35) simply involves looking at most N times in a table of values, N being the number of users in the cell, this is a practically implementable admission rule.

6.3 System Model

We now describe the propagation model used for calculating the mean SINR.

6.3.1 Path Loss

Let $L_{i,s}$ denote the path loss between user i and BS s. We assume that $L_{i,s}$ does not depend on the PRB we are considering, and is given by the following formula:

$$L_{i,s} = A\frac{1}{(d_{i,s})^{\nu}} \tag{36}$$

with $d_{i,s}$ - the distance (in km) between user i and BS s, and A, ν - two constants that depend on the environment.

6.3.2 Shadowing

Let $\chi_{i,s}$ denote the shadowing between user i and BS s. We assume that $\chi_{i,s}$ does not depend on the PRB we are considering, and will be modeled by a log-normal random variable:

$$\chi_{i,s} = 10^{\frac{a\epsilon_1 + b\epsilon_2}{10}} \tag{37}$$

with $\epsilon_i \sim N(0, \sigma^2)$, $i \in \{1, 2\}$ and $a = b = \frac{1}{\sqrt{2}}$.

6.3.3 Interference

We define $\mathcal{S}(i)$ the serving BS for user i, and $\mathcal{N}(i)$ the set of all neighboring BS for user i. We consider neighboring base stations as the only source of interference, and we denote by $P_s^{(k)}$ the power transmitted by BS s on PRB k. Let $I_{i,s}^{(k)}$ denote the interference to user i caused by neighboring BS s on PRB k, which we model by the following:

$$I_{i,s}^{(k)} = P_s^{(k)} A \frac{1}{d_{i,s}^{\nu}} \chi_{i,s} \tag{38}$$

the total interference on PRB k is then:

$$I_i^{(k)} = \sum_{s \in \mathcal{N}(i)} I_{i,s}^{(k)} \tag{39}$$

6.3.4 SINR

We can now calculate the average SINR for user i on PRB k by the following formula:

$$S_i^{(k)} = \frac{P_{\mathcal{S}(i)}^{(k)} \chi_{i,\mathcal{S}(i)} L_{i,\mathcal{S}(i)}}{I_i^{(k)} + {\sigma_N}^2} \tag{40}$$

${\sigma_N}^2$ being the thermal noise.

7. SIMULATION

7.1 Simulator

We implement the coverage-capacity algorithm described above in a realistic OFDMA network simulator with 33 stations to observe its average performance. We use a semi-dynamic network simulator with time resolution of $1s$ (see [12] for a detailed description of a semi-dynamic simulator). For each interval of simulator time, the following operations are performed:

- Computation of the mean throughput of each user
- Calculation of the new positions of mobiles
- Handovers
- Departure of users, due to end of transmission or lack of coverage
- Arrival of users according to a Poisson process and admission control
- Observation of Key Performance Indicators (KPIs) and adjustment of the α

Admission control is done with the algorithm described previously. We consider a streaming service where a user quits the service if he is not covered during 10 consecutive seconds. The number of users that quit the service in such a way is a measure of coverage, and we will show that the proposed algorithm reduces it appreciably.

We compare the proposed algorithm to a reference scenario in which BSs apply a PF scheduler all the time, that is $\alpha_s = 1$, $\forall s$. It is noted that admission control is the same for both algorithms so that the comparison between the proposed algorithm and the reference one, hence the coverage improvement is not related to the admission control strategy.

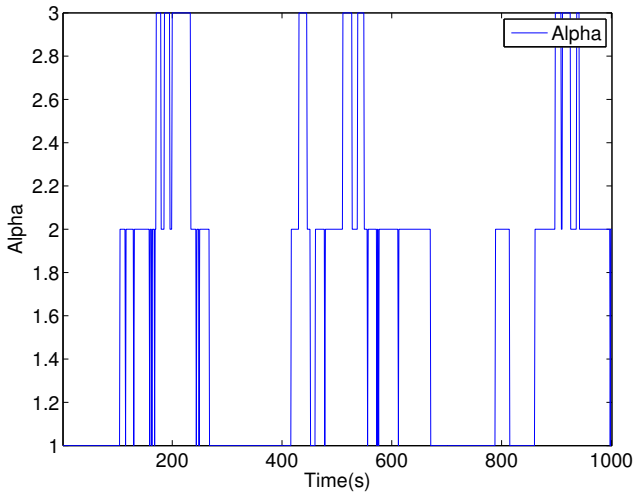

Figure 6: Evolution of α as a function of time for a BS.

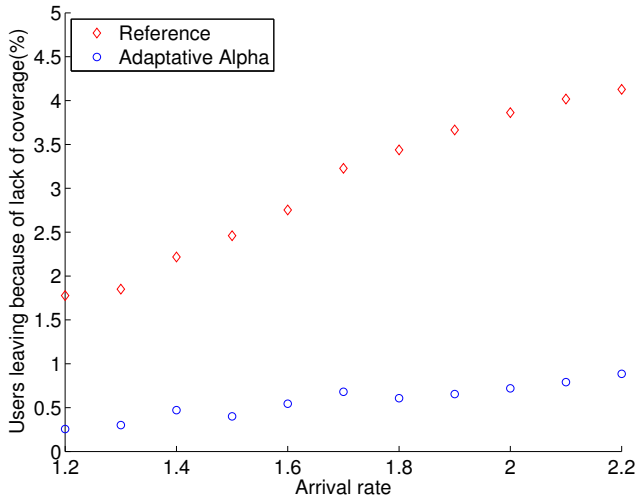

Figure 8: Number of users leaving because of lack of coverage as a function of arrival rate.

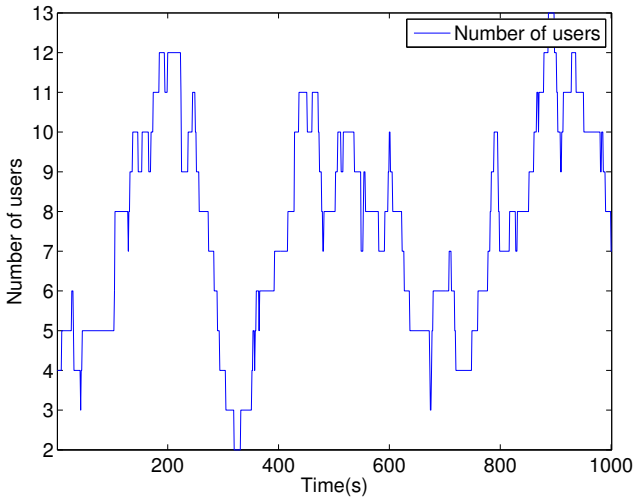

Figure 7: Number of users in a BS as a function of time.

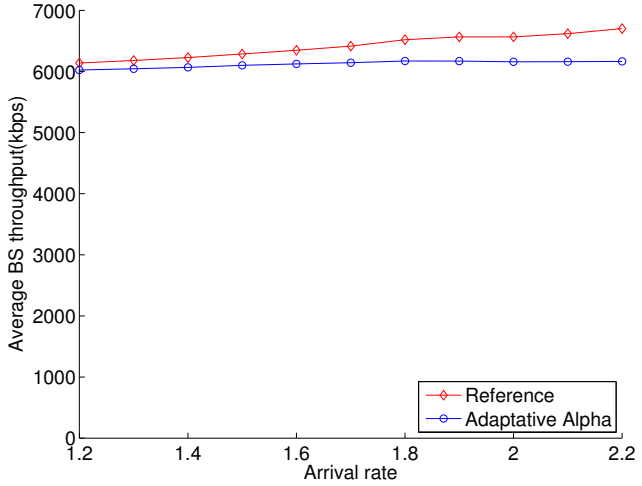

Figure 9: Average BS throughput as a function of arrival rate.

Simulator parameters	
Spatial resolution	$25m \times 25m$
Time resolution	$1s$
Simulation time	$10000s$
User speed	$5km/h$
File length	$120s$
Coverage threshold	$256kb/s$
Network parameters	
Number of PRBs	12
Size of a PRB	$180kHz$
Number of stations	33
Cell layout	11 eNB's $\times$ 3 sectors
Average intercell distance	$1km$
Type of service	Streaming
Propagation	
Thermal noise	-174dBm/Hz
Path loss(d in km)	$128 + 37.6 \log_{10}(d)$ dB
Shadowing standard deviation	6 dB
Antenna configuration	MIMO 2×2

Table 4: Model parameters

7.2 Simulation Results

Figure 6 shows the evolution of α during the simulation for a particular BS, and Figure 7 the number of users served by this BS. We can clearly see that the algorithm keeps α low when the number of users is small, in order not to loose capacity, and increases α when the number of users increases in order to keep all users covered. Figure 8 shows the percentage of users that have left the network because of a lack of coverage, namely because they did not receive the minimal bitrate for 10 consecutive seconds as described above. The proposed algorithm allows to reduce the percentage of users leaving the network from 4% which is generally considered unacceptable in terms of QoS to less than 1%. Figure 9 shows the average BS throughput. The capacity loss caused by the coverage improvement is on average 4%, which is a relatively small price to pay for the important reduction of calls dropped because of coverage loss. It is noted that from a QoS point of view, it is generally much more important to serve more users than to improve the global system throughput.

8. CONCLUSION

This paper has presented a simple and efficient SON algorithm that uses α-fair schedulers to achieve optimal coverage-capacity trade-offs in an OFDMA network. Several formulas for calculating the scheduling gain have been derived based on an approximation of the capacity of a MIMO channel. A Monte-Carlo method for calculating an OFDMA α-fair scheduler throughput is also provided with a proof of convergence. Scheduling gain calculation is necessary in order to implement the algorithm in a network simulator. The algorithm has then been tested on a realistic 33 cells network simulator. Important coverage gains have been achieved at the expense of small capacity losses. The algorithm is scalable and computationally efficient, making it a good candidate for practical implementation.

9. REFERENCES

[1] Requirements for WiMAX Air Interface System Profile Release. Technical Report 2.0, 3rd Generation Partnership Project (3GPP).

[2] 3GPP. Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access (E-UTRAN); Overall description; Stage 2(Release9). TS 36.300, 3rd Generation Partnership Project (3GPP), Apr. 2010.

[3] 3GPP. Telecommunication management; Study on Energy Savings Management (ESM) (Release 10). TR 32.826, 3rd Generation Partnership Project (3GPP), Apr. 2010.

[4] V. S. Borkar. *Stochastic Approximation: A Dynamical Systems Viewpoint*. Cambridge University Press, 2008.

[5] R. Combes, Z. Altman, and E. Altman. On the use of packet scheduling in self-optimization processes: application to coverage-capacity optimization. In *WiOpt 2010*, Avignon, France, June 2010.

[6] M. Dirani and Z. Altman. A cooperative reinforcement learning approach for inter-cell interference coordination in OFDMA cellular networks. In *WiOpt 2010*, June 2010.

[7] W. C. Jakes. *Microwave Mobile Communications*. IEEE Press, 1974.

[8] M. A. Kamath and B. L. Hughes. The asymptotic capacity of multiple-antenna rayleigh-fading channels. *IEEE Transactions on Information Theory*, 51(12):4325–4333, 2005.

[9] H. J. Kushner and G. G. Yin. *Stochastic Approximation and Recursive Algorithms and Applications 2nd edition*. Springer Stochastic Modeling and Applied Probability, 2003.

[10] J. Mo and J. Walrand. Fair end-to-end window based congestion control. *IEEE transactions networking*, 8:556–566, October 2000.

[11] NGMN. NGMN Recommendation on SON and O&M Requirements. Technical report, NGMN Alliance, Dec. 2008.

[12] A. Samhat, Z. Altman, M. Francisco, and B.Fourestie. Semi-dynamic simulator for large scale heterogeneous wireless networks. *International Journal on Mobile Network Design and Innovation (IJMNDI)*, 1(3-4):269–278, 2006.

[13] E. Telatar. Capacity of multi-antenna gaussian channels. *European Transactions on Telecommunications*, 10:585–595, 1999.

[14] D. Tse and P. Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, June 2005.

[15] G. Wunder, M. Kasparick, A. Stolyar, and H. Viswanathan. Self-Organizing Distributed Inter-Cell Beam Coordination in Cellular Networks with Best Effort Traffic. In *WiOpt 2010*, June 2010.