A Unified Scheduling Framework Based on Virtual Timers For Selfish-Policy Shared Spectrum

Alireza Attar
Department of Electrical and Computer engineering
University of British Columbia
E-mail: attar@ece.ubc.ca

Natasha Devroye
Department of Electrical and Computer engineering
University of Illinois at Chicago
E-mail: devroye@ece.uic.edu

Haoming Li and Victor C. M. Leung
Department of Electrical and Computer engineering
University of British Columbia
E-mail: {hlih,vleung}@ece.ubc.ca

***Abstract*—The issue of efficiency and fairness in resource allocation will continue to be of significant importance in scheduler designs for future wireless systems. Of particular importance is the development of *distributed* techniques for achieving desired efficiency and fairness tradeoffs. We will focus on the the design of *selfishly efficient* and as well as different *fair* policies through the use of a virtual timer. This virtual timer unifies various previously considered fairness methods including Round-Robin, Max-Min and Proportional fairness, which are rigorously investigated. The performance of the presented techniques are compared using extensive simulations.**

***Index Terms*—Fairness, Scheduling, Spectrum Sharing, Selfish strategy, Altruistic strategy.**

I. Introduction

As the demand for high bandwidth applications grows, next generation wireless and broadband communications systems strive to achieve highly efficient performance approaching their theoretical limits. From a scheduling perspective, efficient behavior can be obtained by prioritizing nodes with higher channel gains (leveraging multi-user diversity) as is done for instance in opportunistic OFDM-based scheduling schemes. There are, however, two challenges in realizing full efficiency in future wireless technologies. First is the evolution of wireless systems towards more distributed architectures, which shifts the network intelligence towards network edges, i.e., access points. A fitting example is the UMTS Long Term Evolution (LTE) standard, where coordination among evolved NodeBs (eNB) should happen in a distributed manner alleviating the need for a centralized entity generally known as a Radio Network Controller (RNC) in previous cellular systems [1]. Another example from the LTE standard is the support of low-power, autonomously operated access points called Home NodeB (a.k.a. femto-cells in the literature) [2].

The second challenge undermining the importance of purely efficiency-based performance measures in future wireless systems is the need for fairness in resource allocation. Fairness deals with the question of how the resources in the system are distributed amongst users. It is now well understood that efficient scheduling strategies, such as the aforementioned opportunistic scheduling, are highly unfair. This unfairness stems from the fact that only a small proportion of users contributing the most towards the overall system performance are selected for scheduling in efficient resource allocation schemes. On the contrary, a fair scheduler guarantees that almost all users will may access the communication resources (e.g. are scheduled for transmission), no matter how good or bad their channel condition is.

There are many studies in the literature that address efficiency, fairness or both in resource allocation [3]–[6]. In this paper, we propose a *unified framework* for developing distributed resource allocation techniques with different fairness-efficiency tradeoffs. The main contribution of this paper is thus a parameterized virtual timer mechanism which allows the operator to "tune" a distributed network to a specific level of fairness or efficiency by adjusting the parameters of this timer. The rest of this paper is organized as follows. Section II introduces the problem formulation. We focus on efficient resource allocation techniques in Section III and on fairness schemes in Section IV. The simulation results are discussed in Section VI, and we conclude in Section VII.

II. Problem Formulation

Consider a network comprised of K links, whereby a link is defined as a transmitter paired with a receiver. Different links can have a shared transmitter, i.e., a Broadcast Channel (BC), or a shared receiver, such as a Multiple Access Channel (MAC). We assume the operating frequency band is partitioned into N orthogonal resource blocks, where the set of all resource blocks is denoted by $\boldsymbol{\mathcal{R}}$. Further, denote by $\boldsymbol{\mathcal{R}}_i \subseteq \boldsymbol{\mathcal{R}}$ the set of allocated resources to link i, where $i \in \{1, 2, \ldots, K\}$. In this paper we focus on the case where resource blocks are sub-channels and we develop scheduling strategies with mutually exclusive allocated resources similar to OFDMA-based systems, i.e., $\boldsymbol{\mathcal{R}}_i \bigcap \boldsymbol{\mathcal{R}}_{-i} = \varnothing$, where $\boldsymbol{\mathcal{R}}_{-i}$ indicates the set of allocated resources to all other links except link i. ,Note that the developed framework can be easily extended to the case of any orthogonal resource blocks such as time slots or codes. Assume link i has a utility function defined by $\mathcal{U}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right)$. For simplicity of analysis, let us use the achieved throughput as the utility of each link, i.e.,

$$\mathcal{U}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right) = \sum_{n \in \boldsymbol{\mathcal{R}}_i} \log\left(1 + \frac{p_{i,n} g_{i,n}}{\sigma_i^2}\right), \tag{1}$$

where $p_{i,n}$ is the allocated power of link i to sub-channel n, $g_{i,n}$ is the channel gain for that sub-channel and σ_i^2

is the noise power assumed equal in all sub-channels of receiver i. The channel gain is assumed known at both the transmitter and receiver of that particular channel, but not necessarily at any other transmitter or receiver. This channel knowledge mandates a certain amount of signaling overhead for each link, assumed negligible in our analysis. Hence, the presented results can be interpreted as theoretic upper bound for practical scenarios in which channels are known to transmitters and receivers but no other links. We assume the bandwith of sub-channels are narrow enough so that the fading is flat. Finally we assume resource allocation is performed periodically and synchronously in the network. The case of asynchronous networking is not discussed in this paper, due to lack of space. The resource allocation period, denoted by τ is sufficiently short as to justify the assumption that the gain of all sub-channels is constant during each scheduling period.

III. Distributed Selfish Efficient strategy

A greedy approach to access the shared resources is based on a selfish strategy whereby each link strives to maximize its own utility $\mathcal{U}_i(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i})$. The power budget of each link is limitted; this constraint can reflect for example a regulatory demand or a physical limitation of the transmitting device such as its battery life. The overall system performance is measured by sum of the utilities of all links, i.e., the social welfare of the network. Hence, the selfish resource allocation problem can be formulated as

$$\sum_{i=1}^{K} \underset{\boldsymbol{\mathcal{R}}_i}{\text{Maximize}} \sum_{n\in\boldsymbol{\mathcal{R}}_i} \log\left(1+\frac{p_{i,n}g_{i,n}}{\sigma_i^2}\right), \quad (2)$$

subject to

$$\sum_{n\in\boldsymbol{\mathcal{R}}_i} p_{i,n} \le P_{max,i}, \quad \forall\, i\in\{1,2,\ldots,K\}. \quad (3)$$

To solve the above optimization problem in a distributed manner, define the set $\boldsymbol{\mathcal{G}}_i$ as the decreasingly ordered set of channel gains from link i's perspective. The n^{th} element of this set is $\boldsymbol{\mathcal{G}}_i(n) = g_{i,n}$, where $n\in\{1,2,\ldots,N\}$, and it can be inferred that $\boldsymbol{\mathcal{G}}_i(n) > \boldsymbol{\mathcal{G}}_i(n+1)$ due to the ordering of the set. To maximize its own utility, as well as contribute to the maximization of network social welfare in (2), each link prioritizes sub-channels with higher channel gains. Such a system may be modeled using a "virtual timer" mechanism [6] whereby link i accesses sub-channel n after

$$\mathcal{T}_{i,n} = \frac{\mathcal{C}_{self}}{g_{i,n}}, \quad (4)$$

where $\mathcal{C}_{self}$ (S) is an arbitrary constant, selected such that the value of the timers of all links are short enough compared to the rate of change in sub-channel gains. We assume all links will start their timers simultaneously. If after the expiry of timer $\mathcal{T}_{i,n}$, link i discovers that sub-channel n is already occupied by another link, it will wait to access the next sub-channel using timer $\mathcal{T}_{i,n+1}$.

Upon accessing each new sub-channel, link i should determine the distribution of its power such that (3) is satisfied. For a given set of resources, the optimal power allocation for link i can be calculated from the Lagrangian function of optimization problem (2) subject to (3), i.e.,

$$\begin{aligned} L(p_{i,n},\lambda_i) = & \sum_{n\in\boldsymbol{\mathcal{R}}_i} \log\left(1+\frac{P_{i,n}g_{i,n}}{\sigma_i^2}\right) \\ & - \lambda_i\left(\sum_{n\in\boldsymbol{\mathcal{R}}_i} P_{i,n} - P_{max,i}\right), \end{aligned} \quad (5)$$

where λ_i is the Lagrange multiplier. The optimal power allocation is the result of first order (necessary) KKT condition, given by $\frac{\partial L(p_{i,n},\lambda_i)}{\partial p_{i,n}} = 0$. The result is the celebrated water-filing power allocation policy, i.e.,

$$p_{i,n}^{*} = \left(\frac{1}{\lambda_i} - \frac{\sigma_i^2}{g_{i,n}}\right)^{+}, \quad (6)$$

where $(x)^+ = \mathrm{Max}(x,0)$ and the water-level, $\frac{1}{\lambda_i}$, is chosen so that constraint (3) is satisfied. Replacing (6) into (3) yields,

$$\lambda_i = \frac{|\boldsymbol{\mathcal{R}}_i|}{P_{max,i} + \sigma_i^2 \sum_{n\in\boldsymbol{\mathcal{R}}_i} \frac{1}{g_{i,n}}}, \quad (7)$$

where $i\in\{1,2,\ldots,K\}$. Therefore, accessing any additional sub-channel $\hat{n}$ from the set $\boldsymbol{\mathcal{G}}_i$, using (4), increases both the numerator and denominator of (7). Therefore, the marginal improvement of utility value will converge towards a saturation point as $|\boldsymbol{\mathcal{R}}_i| \to N$. In order to limit a given link i from accessing many further sub-channels achieving only infinitesimal improvements in its utility value, and in return, provide the chance for more links to access resources, we consider a further constraint defined by the marginal value of a specific sub-channel $\hat{n}$ by

$$\mathcal{V}(\hat{n}) = \frac{\mathcal{U}_i(|\boldsymbol{\mathcal{R}}_i \bigcup \{\hat{n}\}|) - \mathcal{U}_i(|\boldsymbol{\mathcal{R}}_i|)}{\mathcal{U}_i(|\boldsymbol{\mathcal{R}}_i|)} \ge \psi, \quad (8)$$

where $\hat{n}\in\{1,2,\ldots,N\}$ and $\psi\in\mathbb{Z}^+$ is the cut-off threshold. The new sub-channel $\hat{n}$ should only be accessed if its marginal value in (8) is higher than ψ. In this paper we select $\psi = \%1$. Hence, algorithm 1, executed at the beginning of each resource allocation period, provides a distributed approach to achieve efficiency in a selfish link regime.

IV. Distributed Fairness strategies

In this section, several fair scheduling techniques implemented in a distributed manner using the proposed virtual timer mechanism are introduced.

A. Distributed Max-Fair Policy

As mentioned before the goal of *fair* resource allocation schemes is to guarantee equal (or near equal) distribution of utility over the set of coexisting nodes. Therefore, we can define a Max-Fair (MF) policy as satisfying

$$\mathcal{U}_1(\boldsymbol{\mathcal{R}}_1, \boldsymbol{\mathcal{R}}_{-1}) = \mathcal{U}_2(\boldsymbol{\mathcal{R}}_2, \boldsymbol{\mathcal{R}}_{-2}) = \cdots = \mathcal{U}_K(\boldsymbol{\mathcal{R}}_K, \boldsymbol{\mathcal{R}}_{-K}). \quad (9)$$

Algorithm 1 Distributed Selfish Resource Allocation

Initialization: sort sub-channels in $\boldsymbol{\mathcal{G}}_i$, $\forall i \in \{1,2,\ldots,K\}$,
Start virtual timer $\mathcal{T}_{i,n} = \frac{\mathcal{C}_{self}}{g_{i,n}}$, $\forall i \in \{1,2,\ldots,K\}$ and $\forall n \in \{1,2,\ldots,N\}$,
for n=1 to N **do**
  $i^* = \underset{i}{\operatorname{argmin}}\, \mathcal{T}_{i,n}$,
  Calculate λ_{i^*} using (7), and
  Calculate power allocation using (6), $\forall \hat{n} \in \boldsymbol{\mathcal{R}}_{i^*} \bigcup \{n\}$,
  if $\mathcal{V}(n) \geq \psi$ **then**
    $\boldsymbol{\mathcal{R}}_{i^*} = \boldsymbol{\mathcal{R}}_{i^*} \bigcup \{n\}$,
  else
    $\boldsymbol{\mathcal{R}}^*_{i^*} = \boldsymbol{\mathcal{R}}_{i^*}$, and
    Stop accessing further sub-channel for link i^*,
  end if
end for

Each link needs to select $\boldsymbol{\mathcal{R}}_i$, $\forall i \in \{1,2,\ldots,K\}$ such that

$$\prod_{n\in\boldsymbol{\mathcal{R}}_1}\left(1+\frac{p_{1,n}g_{1,n}}{\sigma_1^2}\right) = \cdots = \prod_{n\in\boldsymbol{\mathcal{R}}_K}\left(1+\frac{p_{K,n}g_{K,n}}{\sigma_K^2}\right). \quad (10)$$

As the utility of all links will be equal, according to (9), the Cumulative Distribution Function (CDF) plot of link utilities normalized by the average link utility of the network will be an straight line with slope 1. This interesting property can be used as a comparison measure of scheduling fairness in practical system, as also suggested by [8].

However, solving (10) requires the knowledge of all sub-channels of all links, which is not feasible in a distributed framework. Further, given the random nature of both the channel gains and the noise power there might be no feasible solution for (10) in a bounded number of sub-channels.

B. Distributed Round-Robin Fair Policy

The RRF policy in a selfish resource allocation regime guarantees allocating a random but equal-size set of resources to each coexisting link, i.e.,

$$\mathbb{E}_t\left\{\left|\boldsymbol{\mathcal{R}}_1(t)\right|\right\} = \mathbb{E}_t\left\{\left|\boldsymbol{\mathcal{R}}_2(t)\right|\right\} = \cdots = \mathbb{E}_t\left\{\left|\boldsymbol{\mathcal{R}}_K(t)\right|\right\}. \quad (11)$$

The averaging process takes place over the time window defined as

$$t \in [t_0 + j, t_0 + j + (m \times \tau)], \quad \text{and } j, m \in \mathbb{N}^+, \quad (12)$$

where t_0 (S) denotes the time origin, j is the resource allocation period index, τ (S) is the length of one resource allocation period and m indicates the number of periods constituting the averaging window length. This fairness strategy can be achieved by scheduling a certain number of links at each resource allocation period, where the number of sub-channels allocated to each scheduled link is given by

$$\left|\boldsymbol{\mathcal{R}}_{inst,i}\right| = \left\lfloor \frac{N}{\mathcal{K}} \right\rfloor, \quad \forall i \in \{1,2,\ldots,K\}, \quad (13)$$

where $\boldsymbol{\mathcal{R}}_{inst,i}$ is the set of allocated resources to link i using RRF policy and $\mathcal{K} \leq K$ denotes the maximum number of simultaneously scheduled links. In practice a simple *average* RRF policy solution allocated all the available resource blocks at each period to one randomly selected link, i.e., $\mathcal{K} = 1$, resulting in an achieved over a window of (12) with $m = K$. A more complex RRF policy is needed to ensure that the averaging window retains its smallest possible value, i.e., $m \to 1$ in (12). This *instantaneous* RRF resource allocation strategy can be implemented by scheduling $\mathcal{K} = K$ links at each resource allocation period. Note that any RRF averaging policy is feasible only if $|\boldsymbol{\mathcal{R}}_{inst,i}| > 0$ in (13).

To implement the RRF resource allocation in a distributed manner, each link should use a random virtual timer equal to $\tau_i = \mathcal{C}^i_{RRF}$, where $i \in \{1,2,\ldots,K\}$, and $\mathcal{C}^i_{RRF}$ (S) is a random variable from a uniform distribution over the interval $[0,\ \mathcal{C}_{max}]$. The value of $\mathcal{C}_{max}$ is chosen such that the longest timer value is short enough compared with the rate of change in sub-channel gains. Depending on the desired period of RRF operation, i.e., *m* in (12), different number of links will be able to access the shared band in each resource allocation period. Each scheduled link will not attempt to access any sub-channel for $m \times \tau$ (S). Each link after expiry of its timer will check the shared band to find out how many links have already been scheduled. This measurement can be based on detecting the unique ID of each link, broadcasted in their occupied sub-channels. The new link will only access idle sub-channels if the number of already scheduled links are at most $N - \left\lfloor \frac{N}{\mathcal{K}} \right\rfloor$.

C. Distributed Max-Min Fair Policy

The optimization problem that models this fairness policy can be written as

$$\underset{\boldsymbol{\mathcal{R}}}{\text{Maximize}}\ \ \underset{i}{\text{Min}}\left(\sum_{n\in\boldsymbol{\mathcal{R}}_i}\frac{B}{N}\log\left(1+\frac{p_{i,n}g_{i,n}}{\sigma_i^2}\right)\right), \quad (14)$$

subject to (3). To solve this problem, recall from Section III that if the coexisting links follow an *efficient* selfish strategy, the links with higher channel gains will access more resource blocks. Therefore those nodes having the lowest channel gains will potentially achieve minimal utility values. The goal of MMF policy is to enhance the performance of such worst-off links at the expense of penalizing links with very high utility value. To this end each link uses a virtual timer value of

$$\mathcal{T}_i = \mathcal{C}_{MMF} \times \widehat{U}_i, \quad \forall i \in \{1,2,\ldots,K\}, \quad (15)$$

where

$$\widehat{U}_i\left(\widehat{\boldsymbol{\mathcal{R}}}_i, \widehat{\boldsymbol{\mathcal{R}}}_{-i}\right) = \sum_{n\in\widehat{\boldsymbol{\mathcal{R}}}_i}\frac{B}{N}\log\left(1+\frac{p_{i,n}g_{i,n}}{\sigma_i^2}\right). \quad (16)$$

In (15), $\mathcal{C}_{MMF}$ (S) is an arbitrary constant to keep the value of $\mathcal{T}_i$ short enough and $\widehat{\boldsymbol{\mathcal{R}}}_i$ in (16) denotes the (hypothetic) optimum set of resource blocks with the highest channel gains for link *i*, *assuming* no other links were competing to access those resource blocks. In that case, $\widehat{U}_i$ denotes the maximum possible utility for link *i* in the absence of coexisting links. Let us define $\underline{i} = \underset{i}{\operatorname{argmin}}\, \widehat{U}_i$. Then the link $\underline{i}$ which has the lowest $\widehat{U}_i$ in the absence of coexisting links, will also have the lowest utility in presence of coexisting links. This intuition is justified because with high probability there exists at least one link with a better channel condition, in at least one of the $\left|\widehat{\boldsymbol{\mathcal{R}}}_{\underline{i}}\right|$ resource blocks of link $\underline{i}$, denying link $\underline{i}$ of accessing that resource block. Hence, by scaling the virtual timer of all links with $\widehat{U}_i$, the links with lower (expected) utility will gain a higher priority in accessing the channels. After expiry of its timer, each link will select the optimal set of resources from the idle sub-channels. Constraint (3) will impose a limit on the total number of accessed sub-channels by each link. Similar to Section III, we assume constraint (8) is enforced. It is straightforward (by contradiction) to verify that the above virtual timer value results in a resource allocation in which the utility of a node cannot be further increased without decreasing the utility of at least another node with already a lower or equal utility value, i.e., a Max-Min Fair equilibrium [3].

D. Distributed Proportional-Fair Policy

Finally, we consider the Proportional-Fair (PF) policy. A PF policy [3], [4] is the result of a resource allocation, $\boldsymbol{\mathcal{R}}_i^*, \forall i \in \{1, 2, \ldots, K\}$, such that for any other resource allocation $\boldsymbol{\mathcal{R}}_i$ we have

$$\sum_{i=1}^{K} \frac{\mathcal{U}_i\left(\boldsymbol{\mathcal{R}}_i^*, \boldsymbol{\mathcal{R}}_{-i}^*\right) - \mathcal{U}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right)}{\mathcal{U}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right)} \leq 0. \qquad (17)$$

In other words, PF policy results in an equilibrium with zero or negative aggregate of proportional changes in the utility of all links in the network using any other resource allocation strategy. A viable solution to reach such an equilibrium state is to use a normalized expected utility perspective [9]. Define $\hat{j}$ as a specific resource allocation period in (12), with a fixed and known m. Define the expected utility for link i in the given resource allocation period $\hat{j}$ by (16). Then, the normalized expected utility of link i at that time slot is defined as

$$\widetilde{\mathcal{U}}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right)|_{t=t_0+\hat{j}} = \frac{\widehat{\mathcal{U}}_i\left(\widehat{\boldsymbol{\mathcal{R}}}_i, \widehat{\boldsymbol{\mathcal{R}}}_{-i}\right)|_{t=t_0+\hat{j}}}{\mathbb{E}_t\left\{\mathcal{U}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right)\right\}}, \qquad (18)$$

where $t \in \left[t_0 + \hat{j} - (m \times \tau),\ t_0 + \hat{j} - 1\right]$. At resource allocation interval $\hat{j}$, links in decreasing order of value of $\widetilde{\mathcal{U}}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right)$ will start accessing the resources, i.e., using the virtual timer

$$\mathcal{T}_i = \frac{\mathcal{C}_{PF}}{\widetilde{\mathcal{U}}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right)}, \quad \forall\ i \in \{1, 2, \ldots, K\}, \qquad (19)$$

where $\mathcal{C}_{PF}$ (S) is an arbitrary constant to scale the value of timers as appropriate. Each link will check availability of sub-channels after expiry of its timer and will follow a selfish resource access procedure, similar to Algorithm 1, if idle sub-channels are available. The average utility in the denominator of (18) will be updated for period $\hat{j}+1$ by [9],

$$\mathbb{E}_{t+1}\left\{\mathcal{U}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right)\right\} = \left(1 - \tfrac{1}{m \times \tau}\right)\mathbb{E}_t\left\{\mathcal{U}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right)\right\} + \tfrac{1}{m \times \tau}\mathcal{U}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right), \qquad (20)$$

where with a slight abuse of notation $\mathbb{E}_{t+1}$ is meant to denote average until $t = t_0 + \hat{j} + 1$.

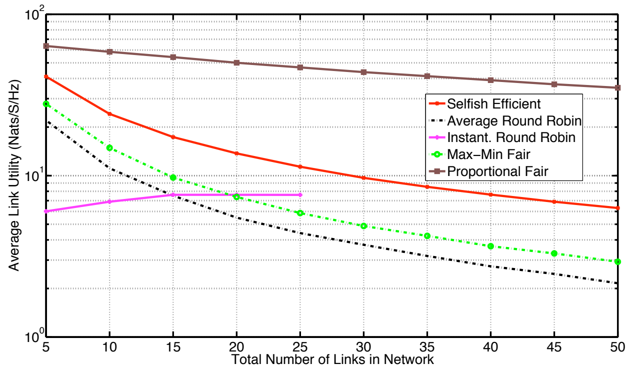

Fig. 1. The average link utility as a function of number of coexisting links.

V. Unified Virtual Timer Design

In the previous sections, we demonstrated the possibility of implementing various efficient or fair selfish policies using a virtual timer mechanism in detail. The significance of the proposed solutions is the flexibility that it will bring about through possibility of using a unified virtual timer with adjustable parameters such that each fair or efficient scheduling technique can be implemented using a software-controlled mechanism rather than dedicated hardware. Such transition of resource allocation strategies is not feasible in the existing systems.

Define the unified virtual timer by

$$\mathcal{T}_{i,n}^{unified} = \mathcal{C}_{arbt}\left(\frac{\beta_1\widehat{\mathcal{U}}\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right) + \beta_2 g_{i,n} + \beta_3}{\beta_4\mathbb{E}_t\left\{\mathcal{U}_i\left(\boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{R}}_{-i}\right)\right\} + \beta_5}\right)^{\alpha}, \qquad (21)$$

where $\mathcal{C}_{arbt}$, α and $\beta_1 - \beta_5$ are the parameters of this timer. It is straight forward to select the aforementioned parameters so that the timer value equals any given policy as described in the previous sections. As an example to achieve selfish efficient scheme, given by (4), we need to select $\mathcal{C}_{arbt} = \mathcal{C}_{self}$, $\beta_1 = \beta_3 = \beta_4 = 0$, $\beta_2 = \beta_5 = 1$ and $\alpha = -1$. Note that if the resource allocation policy does not depend on specific sub-channels but rather on the overall utility of a given link, such as MMF or PF schemes in (15) and (19) respectively, the value of the above timer will be the same for all $n \in \{1, 2, \ldots, N\}$ by setting $\beta_2 = 0$ and selecting all other parameters accordingly.

VI. Numerical Results

In this section the performance of the proposed distributed scheduling schemes are investigated via extensive simulation results. We have considered a square area of 1 Km×1 Km dimension. Coexisting links were created using a uniformly random distribution over the considered space. Simulations were repeated 1500 times to obtain average performance results. The maximum transmission power of each link is assumed 200 mW. The channel pathloss exponent is assumed to be 3, with 3 dB log-normal shadowing and Rayleigh-distributed fading with unit mean. We have assumed existence of 25 sub-channels.

Fig. 1 represents the average link utility using the developed distributed resource allocation techniques in Sections III

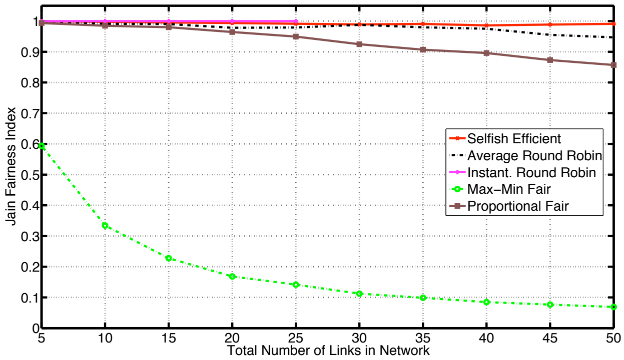

Fig. 2. Jain's fairness index as a function of number of coexisting links.

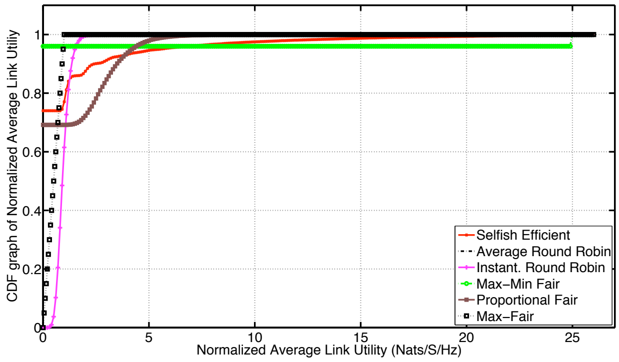

Fig. 3. The CDF graph of normalized average link utility for $K = 25$.

and IV. As evident from this figure, the lowest level of efficiency belongs to instantaneous RRF policy. This result comes as no surprise, given the high level of fairness in instantaneous RRF policy. Given the random nature of sub-channel allocation in RRF regime some links will experience sever fading channels and hence achieve a low level of utility. Further, as the total number of sub-channels allocated to each link is very low, i.e., $|\mathcal{R}_i| \rightarrow 1, \forall\ i \in \{1, 2, \ldots, K\}$, the overall network utility will be very low too. However, as the total number of links in the network increases, instantaneous RRF will outperform average RRF and MMF.

The scheduling schemes that are biased towards efficiency sustain a higher level of average throughput compared to fair scheduling techniques. An interesting outcome of our numerical analysis is that the distributed PF policy can outperform the purely selfish method. Recall from previous sections that unlike centralized scheduling scheme, there is no guarantee that each link can access its optimal set of resources. Therefore, by incorporating a degree of fairness, as the distributed PF policy demonstrates, the overall performance of the network will be improved.

A classic fairness measure is the Jain's fairness index [10]. In Fig 2 Jain's fairness index as a function of number of coexisting links is depicted. Instantaneous RRF policy achieves a high level of fairness, due to scheduling all coexisting links in any given time slot (obviously only applicable if $|\mathcal{R}| \leq K$). Note that PF scheduler will achieve a lower level of fairness compared to RRF, as the tradeoff between efficiency and fairness is more inclined towards efficiency than in RRF scheme. Jain's fairness index provides only an *average* view of fairness of the proposed schemes. A more accurate approach is through the CDF graph of the normalized link throughput by the average network throughput. As discussed in Section IV-A, the benchmark here is the MF scheduler which allocates resources such that all links will achieve an equal utility value as shown in Fig. 3. Again, instantaneous RRF exhibits the most resemblance to the benchmark fairness scheme, i.e., MF policy. Also PF manifests a more evenly distributed utility value amongst the coexisting links.

VII. Conclusions

In this paper we developed a framework to design distributed resource allocation techniques pertinent to the future wireless technologies. To this end, we have used the concept of virtual timer as a convenient approach to coordinate resource access of a distributed network without the need of a centralized decision maker. We propose a model for the virtual timer which encompasses a number of previously considered fairness schemes in terms of seven key tunable parameters. This scheme is a first step towards a fully adjustable scheduling technique whereby the network can switch, in a distributed manner, between various resource allocation efficiency and fairness targets. As of our future work, we will more rigorously study the performance of the proposed virtual timer mechanism, for instance to quantify the performance loss due to asynchronous scheduling in the network. Further, we will study other fairness and efficient regimes such as altruistic approaches.

References

[1] 3GPP technical Specification Group Radio Access Network, "E-UTRA and E-UTRAN Overall description, "*3GPP TS 36.300 V9.0.0*, Jun. 2009.

[2] 3GPP technical Specification Group Radio Access Network, "UTRAN architecture for 3G Home NodeB," *3GPP TS 25.467 V8.1.0*, Mar. 2009.

[3] F. Kelly, "Charging and rate control for elastic traffic," *European Trans. on Telecomm.*, Vol. 8, pp. 33-37, 1997.

[4] F. Kelly and A. Maulloo and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, Vol. 49, No. 3, pp. 237-252, 1998.

[5] 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, March 2003.

[6] H. Seo and B. G. Lee, "Proportional-fair power allocation with CDF-based scheduling for fair and efficient multiuser OFDM systems," *IEEE Trans. on Wireless Commun.*, Vol. 5, No. 5, pp. 978983, May 2006.

[7] D. Qiao, S. Choi and K. G. Shin, "Goodput Analysis and Link Adaptation for IEEE 802.11a Wireless LANs," *IEEE Trans. on Mobile Computing*, Vol. 1, No. 4, pp. 278-292, Oct.-Dec. 2002.

[8] R. Irmer, et. al (editor), "NGMN Radio Access Performance Evaluation Methodology," *NGMN white paper*, Jan. 2008.

[9] P. Viswanath, D. Tse and R. Laroia, " Opportunistic beamforming using dumb antennas," *IEEE Trans. Info. Theory*, Vol. 48, No. 6, pp. 1277-1294, Jun. 2002.

[10] R. Jain, D. M. Chiu and W. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Systems," *DEC Research Report TR-301*, 1984.