Packet Scheduling and Resource Allocation for Downlink Multicarrier Systems with Quality-of-Service Constraints

Qing Bai, Yao Hao and Josef A. Nossek
Institute for Circuit Theory and Signal Processing
Technische Universität München, Munich, Germany
Email: bai.qing@nws.ei.tum.de

***Abstract*—We consider the scheduling and resource allocation problem in the downlink of multicarrier systems where data is processed and transmitted in unit of packets. This involves the allocation of transmit power as well as time and frequency slots to packets generated by different service flows, which consequently have various lengths and allows for various latency time in delivery. An optimization to maximize system throughput under frequency division multiple access (FDMA) and available resources restrictions is formulated, and an interactive scheduling and resource allocation approach is proposed to solve this combinatorial-natured problem. The paper especially focuses on the design of packet scheduling algorithms and introduces the concept of *virtual packet size* and *anxious scheduler*, which allows for simple implementation and high flexibility. Simulation results show the efficient collaboration of the proposed scheduling algorithm with the resource allocation scheme, and the effectiveness of the system as a whole which favorably exhibits low complexity.**

I. INTRODUCTION

For future wireless communication systems which are expected to support various high data rate services and become more energy efficient, scheduling and resource allocation is a critical issue which has been considered in many works using a cross-layer methodology [1]–[3]. Multiuser multicarrier systems, *e.g.*, *orthogonal frequency division multiple access* (OFDMA), provide high flexibility in scheduling and resource allocation but pose at the same time the challenge of finding efficient algorithms with low complexity that optimize system performance and/or provide satisfactory *quality of service* (QoS). From a cross-layer design perspective, channel-aware scheduling is known to benefit system performance by exploiting *multiuser diversity*. Gradient-based scheduling and resource allocation which attempts to maximize the projection of a rate vector onto the gradient of a system utility function provide a generalized optimization framework for different scenarios and applications [4], [5]. These works, as well as many other existing literature on the topic, are based on an information theoretic model where the physical layer is characterized by the achievable rate region obtained with the Shannon formula. Our work takes a divergent approach by employing the cutoff rate theorem and including packet retransmission protocols in the cross-layer system model, which enables us to investigate the scheduling and resource allocation problem for discrete, packetized data.

The main contribution of this work lies in the design, implementation and simulation of several new scheduling methods as compared to our previous work [6]. The improvement in scheduler design and its collaboration with the resource allocation module results in an overall improvement in system performance, measured by long-term throughput. The rest of the paper starts with a detailed explanation of the system structure and module functionalities in Section II, followed by a description of the resource allocation algorithm employed by the resource allocator in Section III. The scheduling algorithms will be proposed and discussed in Section IV. Simulation results are shown and analyzed in Section V before the paper is summarized and concluded in Section VI.

II. SYSTEM STRUCTURE AND FUNCTIONALITY

Consider the scenario of an isolated cell in which K users, each with one service flow, are to be served by the base station (BS) which employs a multicarrier system. The BS and the mobile stations are all equipped with single antenna. Due to the different characteristics of the applications that generate the traffic, the services flows have different QoS requirements in minimum reserved data rate, delay tolerance, *etc*. We consider in this work four service flow types as defined in the IEEE 802.16 standard [7]: *Unsolicited Grant Service* (UGS), *Real-time Polling Service* (rtPS), *Non-real-time Polling Service* (nrtPS) and *Best Effort* (BE) service. Their relevant specifications and properties are listed in Table I, where $D_{\min}$, $D_{\max}$, $L_{\min}$ and $L_{\max}$ are service flow dependent parameters.

Table I: Service Flow Types and Traffic Properties

Type	Data rate	Packet size	Max. latency	Packet generation
UGS	constant	constant	finite const.	periodic
rtPS	$[D_{\min}, D_{\max}]$	$[L_{\min}, L_{\max}]$	finite const.	periodic
nrtPS	$[D_{\min}, D_{\max}]$	$[L_{\min}, L_{\max}]$	none	non periodic
BE	$[0, D_{\max}]$	$[L_{\min}, L_{\max}]$	none	non periodic

At each *Transmission Time Interval* (TTI), which is the processing and transmission unit in time, there are a group of packets from the K users stored in the buffer of the BS. From a scheduling and resource allocation point of view, each packet is characterized by its size and latency requirement measured in Bytes and ms, respectively. Let the set of packets in the system buffer at the i-th TTI be $\mathcal{L}_i$ with $|\mathcal{L}_i| = L_i$. Due to the limitation on available resources such as bandwidth and transmit power, the system is not always able to serve every packet from $\mathcal{L}_i$. Denote the set of packets that are transmitted in the i-th TTI by $\mathcal{R}_i$, which gives $\mathcal{R}_i \subseteq \mathcal{L}_i$. Note that some packets from $\mathcal{R}_i$ might not be successfully decoded by the end users, depending on the receive signal-to-noise ratio.

Our long-term goal is to maximize the total amount of data that are successfully received by the end users for a large number of TTI's. On the one hand, if we consider only one particular TTI, to maximize the amount of data that can be successfully served, *i.e.*, to solve

$$\max_{\mathcal{R}_i \subseteq \mathcal{L}_i} \sum_{l \in \mathcal{R}_i} s_l \cdot \lambda_l \tag{1}$$

where s_l is the size of the l-th packet in $\mathcal{R}_i$ and λ_l is the expected probability that the packet transmission can be successful, is intuitively helpful in achieving a large long-term throughput. However, this is a complicated task due to two reasons. First, enumerating all subsets of $\mathcal{L}_i$ is practically impossible for large L_i. Second, to determine whether there are enough resources to serve a subset of packets is complicated itself for multicarrier systems, and often we need to settle for suboptimal resource allocation solutions. On the other hand, solving (1) independently for a large number of TTI's is not identical to maximizing the system throughput, for the packets have various latency requirements and once the deadline for receiving has passed, the packet becomes obsolete and is deleted from the buffer. This highlights the need for a smart scheduling algorithm, which makes short-term decisions for each TTI but serves the long-term goal of maximizing throughput. To this end, we propose an efficient mechanism to come off the aforementioned difficulties [6], where the addressed scheduling and resource allocation problem at the downlink of multiuser multicarrier systems is solved by two separate, yet interactive modules, *i.e.*, the *resource allocator* (RA) working across the PHY and MAC layers, and the *scheduler* which works on top of RA. For the i-th TTI, the functionality of the scheduler is to prioritize the packets in $\mathcal{L}_i$ and then pass the prioritized packet list to the RA. The RA then looks for the specific subchannel assignment, power allocation and modulation and coding schemes (MCS) that could serve the list of packets best, given the channel realization during the i-th TTI. The resource allocation strategy and the result of transmission are fed back to the scheduler for packet management, *e.g.*, insertion and depletion of packets to or from the buffer. A diagram of the system structure and component interactions is given in Figure 1.

Earliest Deadline First (EDF) is a common scheduling policy that takes the maximal latency of packets into account. Our discussion in Section IV and the simulation results in Section V show that treating the packets in an anxious way outperforms EDF and some other related strategies. The RA on the other hand, is responsible for finding a good solution to (1) with reasonable complexity, given a priority order of the packets. To this end, we propose to maximize the number

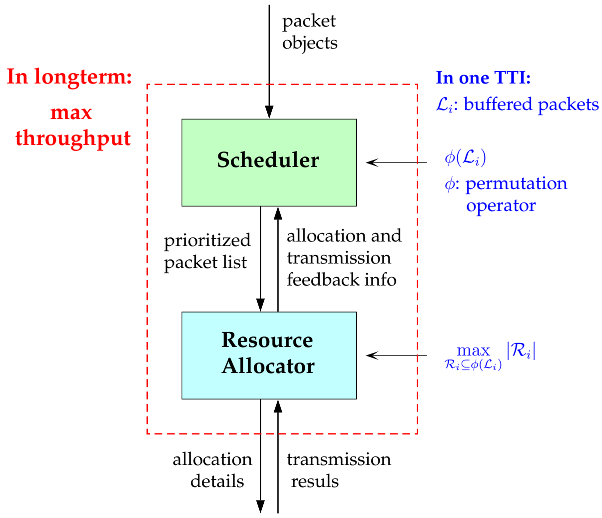


Figure 1: System Structure and Function Description

of packets that can be transmitted, *i.e.*, $\max |\mathcal{R}_i|$, based on the priority order. We have investigated how to compute the minimal transmit power required to serve a set of packets in [8], and a resource allocation algorithm which fits our need here will be described in Section III. The algorithm can be used to test whether a given subset of $\mathcal{L}_i$ is servable, where the test subsets $\mathcal{R}_i$ are generated systematically with Algorithm 1 using the priority order [6]. The term P_{tot} therein denotes the total transmit power that is available at the BS, and ϕ represents the permutation operation done at the scheduler. Note that $\mathcal{R}_i$ does not always need to be initialized as $\mathcal{L}_i$. A proper initialization of $\mathcal{R}_i$ can effectively reduce the number of calls of the power minimization algorithm.

Algorithm 1 Generation of $\mathcal{R}_i$

initialize $\mathcal{R}_i \leftarrow \mathcal{L}_i$, $l \leftarrow L_i$
$P_{\min} \leftarrow$ minimum transmit power to serve $\mathcal{R}_i$
while $P_{\min} > P_{\text{tot}}$ **do**
  delete from $\mathcal{R}_i$ the packet indexed $\phi(l)$, $l \leftarrow l-1$
  $P_{\min} \leftarrow$ minimum transmit power to serve $\mathcal{R}_i$
end while
if $l < L_i - 1$ **then**
  $l \leftarrow l+2$
  while $l \leq L_i$ **do**
    add to $\mathcal{R}_i$ the packet indexed $\phi(l)$, $l \leftarrow l+1$
    $P_{\min} \leftarrow$ minimum transmit power to serve $\mathcal{R}_i$
    if $P_{\min} > P_{\text{tot}}$ **then**
      delete from $\mathcal{R}_i$ the packet indexed $\phi(l-1)$
    end if
  end while
end if

III. Cross-layer Model and Resource Allocation Algorithm

Before coming to the formulation of the transmit power minimization and the algorithm part, we shortly introduce the cross-layer system model which stems from [9] and lays the basis for our optimization.

A. Cross-layer System Model

Definition 1: The QoS parameter *latency* τ of a packet is defined as the delay it experiences until received correctly with an outage probability of no more than the predefined value $\pi^{(\mathrm{out})}$. Let $f[m]$ be the probability that it takes exactly m TTI's to transmit a packet error-free, then

$$\tau = (M-1)(T_{\mathrm{R}} + T_{\mathrm{I}}) + T_{\mathrm{I}},$$

where T_{R} represents the *round trip delay*, T_{I} stands for the length of an TTI, and

$$M = \min_{M'} M' \quad \text{s.t.} \quad \sum_{m=1}^{M'} f[m] \geq 1 - \pi^{(\mathrm{out})}.$$

1) Channel Model: The wireless channel is modeled as frequency-selective fading over its whole bandwidth and frequency-flat fading over each *subchannel*, which consists of N_{c} adjacent subcarriers. Assuming that one TTI contains N_{s} symbols for data transmission, we define the *minimum allocation unit* (MAU) as an indivisible allocation region of one subchannel in the frequency dimension by one TTI in the time dimension, which contains $N_{\mathrm{c}}N_{\mathrm{s}}$ symbols. Perfect channel state information is assumed at the transmitter (CSIT). On a particular MAU n, let $H_{k,n}$ and $\sigma^2_{k,n}$ be the channel coefficient and the Gaussian noise variance of user k, and p_n be the amount of power being allocated to the MAU. When assigned to a packet of user k, the *signal-to-noise-ratio* (SNR) is computed as $\gamma_{k,n} = \frac{|H_{k,n}|^2}{\sigma^2_{k,n}} \cdot p_n$. We drop the subscripts k and n when there is no ambiguity in the following subsections.

2) FEC coding and modulation: With reference to the WiMAX standard, 8 MCS are chosen as candidates to be employed by the MAU's as listed in Table II, and they form a set of *modes of operations* denoted by $\mathcal{M}$.

Table II: Modulation and Coding Schemes (MCS)

Index	Modulation Type	Alphabet Size A	Code Rate R	$R\log_2 A$
1	BPSK	2	1/2	0.5
2	QPSK	4	1/2	1
3	QPSK	4	3/4	1.5
4	16-QAM	16	1/2	2
5	16-QAM	16	3/4	3
6	64-QAM	64	2/3	4
7	64-QAM	64	3/4	4.5
8	64-QAM	64	5/6	5

Let the modulation alphabet and coding rate on the MAU under consideration be $\mathcal{A} = \{a_1, \ldots, a_A\}$ and R respectively. The noisy channel coding theorem [10] states that there always exists a block code with block length l and binary code rate $R\log_2 A \leq R_0(\gamma, A)$ where $R_0(\gamma, A)$ represents the *cutoff rate*, such that with maximum likelihood decoding the error probability $\tilde{\pi}$ of a code word satisfies $\tilde{\pi} \leq 2^{-l(R_0(\gamma,A) - R\log_2 A)}$. This upper bound is applied to the extensively employed turbo decoded convolutional code by using the *equivalent block length* $n_{\mathrm{eq}} = \beta \ln L$, where the parameter β is used to adapt this model to the specifics of the employed turbo code, and L is the length of coded information data [9]. Consequently, the transmission of L bits is equivalent to the sequential transmission of L/n_{eq} blocks of length n_{eq} and has an error probability of

$$\pi = 1 - (1-\tilde{\pi})^{\frac{L}{n_{\mathrm{eq}}}} \leq 1 - \left(1 - 2^{-n_{\mathrm{eq}}(R_0(\gamma,A) - R\log_2 A)}\right)^{\frac{L}{n_{\mathrm{eq}}}}.$$

3) Retransmission protocol: At the link layer the *automatic repeat request* (ARQ) protocol is employed. The data sequence transmitted in one MAU, which will be referred to as a *subpacket*, is used as the retransmission unit since it is independently decodable. A limit $\tilde{m}$ is set on the maximum number of transmissions allowed. With ARQ protocol the corrupted subpackets at the receiver are simply discarded, hence we assume that the error probability of a retransmitted subpacket is the same as that of its original transmission, *i.e.*, $f[m] = \pi^{m-1}(1-\pi)$, where $m \in \mathbb{Z}^+$, $\pi = \pi[1]$.

Denote the set of MAU's assigned to packet l as $\mathcal{P}_l$, and the number of information bits of packet l loaded on MAU n as $B_{l,n}$. The complete transmission of packet l requires $\sum_{n\in\mathcal{P}_l} B_{l,n} = s_l$. The latency time τ_l is determined by the largest subpacket error probability of packet l, denoted by π_l which is given as $\max_{n\in\mathcal{P}_l} \pi_{l,n}$.

The system parameters are summarized in Table III, including some of their notations and the values used for simulations.

Table III: System Parameters

Total bandwidth		10 MHz
Center frequency	f_{c}	2.5 GHz
FFT size		1024
Number of data subcarriers		720
Number of subchannels	N	
Number of subcarriers per subchannel	N_{c}	
Transmission Time Interval (TTI)	T_{I}	2 ms
Symbol duration	T_{s}	
Number of data symbols per TTI	N_{s}	16
Round Trip Delay (RTD)	T_{R}	10 ms
Maximum number of transmissions allowed	$\tilde{m}$	5
Turbo code dependent parameter	β	32
Outage probability	$\pi^{(\mathrm{out})}$	0.01

B. Resource Allocation Algorithm

We formulate the transmit power minimization problem for a given set of packets $\mathcal{R}$ with $|\mathcal{R}| = L$ as

$$\begin{aligned} \min_{\boldsymbol{B}\in\mathcal{B}} \quad & \sum_{l=1}^{L}\sum_{n=1}^{N} \varphi_{k,n}(B_{l,n}, \tau_l^{(\mathrm{rq})}) \\ \text{s.t.} \quad & \sum_{n=1}^{N} B_{l,n} = s_l, \quad l = 1, \ldots, L, \end{aligned} \tag{2}$$

where $\boldsymbol{B} \in \mathbb{Z}_{+,0}^{L \times N}$ represents the bit-loading matrix and $\mathcal{B} \subset \mathbb{Z}_{+,0}^{L \times N}$ stands for the set of matrices that have only one nonzero entry in each of their columns. The complete transmission of each packet is guaranteed by the equality constraint, and the latency requirements are contained in function $\varphi(B, \tau^{(\mathrm{rq})})$, which is formally defined as the minimum power required for the successful transmission of B bits within latency $\tau^{(\mathrm{rq})}$, *i.e.*,

$$\varphi(B, \tau^{(\mathrm{rq})}) \triangleq \min_{(A,R) \in \mathcal{M}} \left\lceil \frac{s}{N_s} \right\rceil \cdot \gamma(B, A, R) \cdot \frac{\sigma^2}{|H|^2}, \tag{3}$$

where $\gamma(B, A, R)$ is the SNR required to convey B bits with maximal $\left\lfloor \frac{\tau^{(\mathrm{rq})} - T_{\mathrm{I}}}{T_{\mathrm{R}} + T_{\mathrm{I}}} + 1 \right\rfloor$ transmissions when the MCS (A, R) is employed, which can be obtained from a binary search on the cutoff rate curve, and $s = \left\lceil \frac{B}{R \log_2 A} \right\rceil$ is the number of symbols occupied in the MAU. Note that the minimization in (3) is independent of the channel realization, hence the optimal MCS corresponding to each B and each maximal number of transmissions can be computed offline.

Due to the exclusive assignment of MAU's and the discrete MCS levels that are available, the transmit power minimization problem is combinatorial and computationally intractable when the number of MAU's and the number of packets are large. Therefore we proposed a suboptimal algorithm of low complexity based on the methods described in [8].

1) The stepwise approach: The minimization in (2) is solved in three steps: MAU assignment, bit and power allocation (BPA), and adjustment. In the first step, the number of MAU's assigned to each packet is first determined, and then the same MCS is assumed for each packet according to this number. The sum power minimization based on these assumptions can be solved by relaxing the integer constraints and then employing linear programming. With the result of the first step, problem (2) is no longer coupled among the packets, and L independent power minimizations can be solved by using Lagrangian dual methods. Since the objective function is nonconvex, some heuristic method is needed to recover a primal feasible bit-loading from the dual optimal solution. The outcome of BPA might indicate zero MCS on some MAU's, which means these MAU's are released from occupation and can be assigned to other packets. As higher MCS are more power consuming, we find the MAU's using the relatively highest MCS and their possessors, and compare each alternative of assigning the empty MAU's to these packets.

2) Adaptive subchannel size: As one MAU can be assigned to at most one packet, the number of MAU's is practically a hard limit on the number of packets that the system could serve in one TTI. However, an MAU may be under utilized if the loaded subpacket is small. In order to better adapt to diverse traffic situations, we make the number of subcarriers that make up one subchannel, *i.e.*, N_{c}, an optimization variable. Since the channel is assumed constant over each MAU, the bandwidth of one subchannel should not exceed the coherence bandwidth of the multipath channel. On the other hand, it is unnecessary and impractical to have a very high frequency resolution due to the usage of pilot subcarriers and the increase in control overhead. As a result, upper and lower bounds on N_{c} and an appropriate search interval can be set, which enable an exhaustive search for the optimal subchannel size.

IV. Scheduling Algorithm

Recall that in the scheduling and resource allocation module we consider, a prioritized packet list is expected from the scheduler in each TTI. In [6], the priorities of the packets are determined first by their service flow types, and then by the scheduling methods used for each service flow type. To be more specific, packets generated by the four service flow types strictly follow the priority order $\mathrm{UGS} > \mathrm{rtPS} > \mathrm{nrtPS} > \mathrm{BE}$. Within each service flow type, packets are sorted according to a certain criterion, *e.g.*, UGS packets are scheduled in a round-robin fashion, and rtPS packets obey the EDF principle. Despite its simplicity, this design is towards achieving a minimized packet loss rate and is rather lack of flexibility to incorporate other factors that might influence the scheduling decision, *e.g.*, unequal weights of the users. To this end, we are motivated to explore new scheduling methods that aims at improving the system throughput.

From Algorithm 1 it is clear that packets of higher priorities are more likely to get served. Thus it is intuitive to schedule the packets according to the descending order of their sizes, when the served data amount is to be maximized. This policy is termed as *largest packet first* (LPF). We propose a so-called *virtual packet size* model to extend and enrich this idea.

A. Virtual Packet Size

The virtual packet size is a virtual attribute assigned to a packet which helps in assorting the packets with a desired emphasis. It has the flexibility to include various factors into the model. For example, if the users in the system has unequal weights, the virtual packet size can be defined as the real packet size times the weight of the corresponding user. Sorting the packets according to their virtual packet sizes leads to the *largest virtual packet first* (LVPF) policy. In the example, this would result in giving higher priorities to the packets from users with larger weights, which provides them with better chances to get system resources.

When latency requirements are to be considered, the virtual packet size can be defined as the real packet size divided by the maximal number of transmissions allowed. Let the real size of a packet be l. If its latency requirement allows it to be transmitted at most M times, then its virtual packet size v is

$$v = \frac{l}{M}. \tag{4}$$

The consideration behind (4) is that the packet can be thought of as being equally split into M parts and transmitted in M TTI's. The larger the latency a packet can tolerate, the smaller its virtual packet size. The scheduling of packets in descending order of v as defined in (4) is referred to as *LVPF 1*.

Channel coefficients can also be used in producing a virtual packet size. In the multicarrier setting, we average the channel gain across all subchannels to have a single weighting factor which can be used to scale the real packet size. Since the difference between channel gains of the users can be much bigger than the difference in packet sizes, we perform a logarithmic operation on the average channel gain before it is multiplied to the real packet size to avoid overemphasizing channel conditions. Let α be the average channel gain of the user which is to receive the packet, then

$$v = l \cdot \log(1+\alpha). \tag{5}$$

Scheduling packets according to the descending order of v as defined in (5) is referred to as *LVPF 2*. Moreover, a combination of the two considerations yields another possibility of packet scheduling, *i.e.*, with virtual packet size

$$v = \frac{l \cdot \log(1+\alpha)}{M}, \tag{6}$$

which will be denoted as *LVPF 1+2*.

B. Anxious Scheduler

Energy efficiency issues involved with retransmission protocols have been discussed in [11], where we claim that always allowing for the maximal number of transmissions of a packet is in general energy inefficient. A transmit power constrained minimization of energy expenditure for the successful delivery of a list of packets is proposed to account for the trade-off between transmit power used in the current TTI and the energy consumption over the whole delivery process. When viewed from long-term, another problem with packet retransmission can be observed as they take extra time/frequency slots from the system which prevents other packets from being transmitted. To this end, an "*anxious*" scheduling principle can be employed by the scheduler, which screens latency information from the RA such that the RA treats all packets as if they have only one chance to be transmitted. This scheduling method can be combined with LPF and LVPF 2 polices, creating two more scheduling options, namely *A-LPF* and *A-LVPF 2*.

V. Simulation Results

To observe the long-term system behavior more accurately, we set up a service flow table which contains the QoS parameters of 28 designed service flows, 8 each for UGS, rtPS and nrtPS flow types, and 4 for BE services. In each simulation, the service flow types and parameters of the K users are chosen from the table, and user locations in a cell of radius 1.5 km are uniformly generated. One simulation lasts for 500 consecutive TTI's which counts for 1 second. The wireless channel is modeled as a frequency-selective fading channel consisting of 6 independent Rayleigh multipaths with an exponentially decaying power profile where the delay spread is 1 μs. The path loss in dB is computed as $PL(d) = 140.6 + 35.0 \log_{10} d$ following the COST-Hata model, where d is the distance between MS and BS in km, and all receiver noise levels are -174 dBm/Hz.

We mainly use the metric *service ratio* to compare the performance of different scheduling algorithms, which is the ratio between the amount of data that has been successfully transmitted by the end of one simulation to the amount of data that has been generated and put into the buffer of the system. With identical traffic input to all schedulers, evaluating service ratio is equivalent to evaluating the throughput. In the first test, we set $K = 16$ and simulate all proposed scheduling algorithms including EDF. The overall service ratio after 40 simulations of EDF, LVPF-1 and LVPF 1+2 are: 0.9263, 0.9515 and 0.9619, respectively. Service ratio for the other scheduling methods are listed in Table IV for an easy comparison with the second test case where $K = 20$. A significant gain of employing anxious schedulers can be read immediately, and the combination with virtual packet size scheme 2 which takes channel conditions into account provides an even larger gain. Detailed service ratios from each simulation, as depicted in Figure 2 and Figure 3, show that the performance comparisons between different scheduling algorithms can be indecisive, except for the anxious schedulers which are constantly better than the others. Depending on the specific user distribution and service flow parameters, one scheduler design might outperform another, but when the scenario changes, the contrary may become true. Hence, if we have enough a priori information about the user and traffic situation in the system, we can choose the most suitable scheduling algorithm or even tune the parameters for computing the virtual packet size; otherwise, the scheduler that performs the best on average, *e.g.*, A-LPF or A-LVPF 2, can be employed.

Table IV: Overall Service Ratio

	LPF	LVPF 2	A-LPF	A-LVPF 2
$K = 16$	0.9338	0.9570	0.9975	0.9985
$K = 20$	0.8556	0.9138	0.9852	0.9896

VI. Conclusion

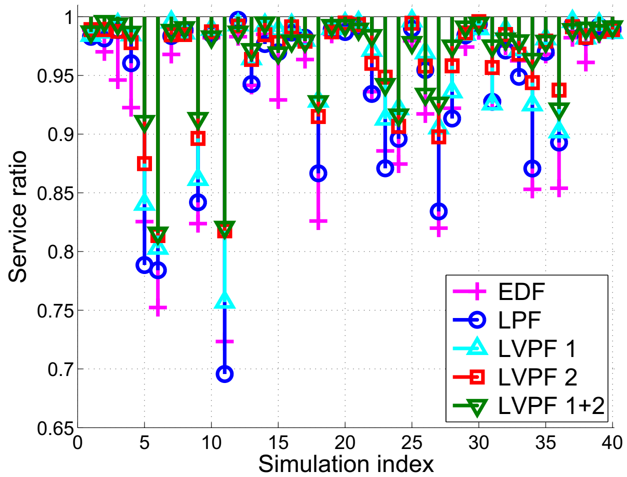
We have proposed an interactive mechanism to efficiently solve the scheduling and resource allocation problem at the downlink of a multicarrier system with multiple end users. With the introduction of the concepts of virtual packet size and anxious scheduler, new designs of packet scheduling algorithms have approached the goal of improving long-term system throughput under heterogeneous QoS requirements, as well as providing more flexibility in incorporating other decision factors. Simulation results demonstrate the superior performance of the proposed algorithms with different traffic densities and characteristics.

References

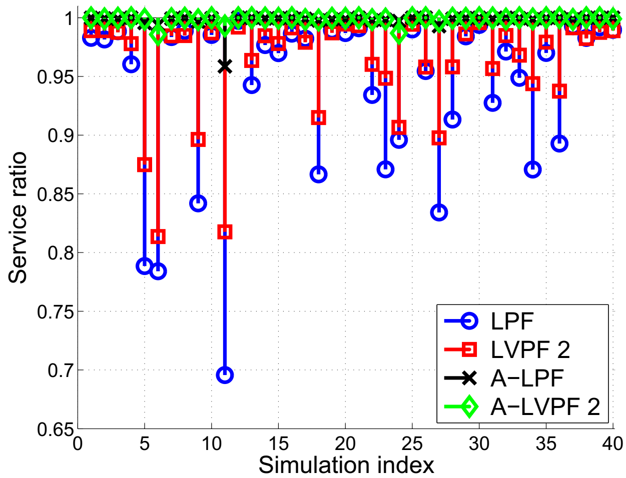
[1] S. S. Jeong, D. G. Jeong, W. S. Jeon, *Cross-layer Design of Packet Scheduling and Resource Allocation in OFDMA Wireless Multimedia Networks*, IEEE 63rd Vehicular Technology Conference, 2006-Spring, pp. 309-313, May 2006.

[2] G. Song, Y. Li, L. J. Cimini, H. Zheng, *Joint channel-aware and queue-aware data scheduling in multiple shared wireless channels*, 2004 IEEE Wireless Communications and Networking Conference, vol. 3, pp. 1939-1944, March 2004.

[3] G. Song, Y. Li, *Utility-based resource allocation and scheduling in OFDM-based wireless broadband networks*, IEEE Communications Magazine, vol. 43, issue 12, pp. 127-134, Dec. 2005.



(a) without anxious scheduler



(b) with anxious scheduler

Figure 2: Service ratio, $K = 16$

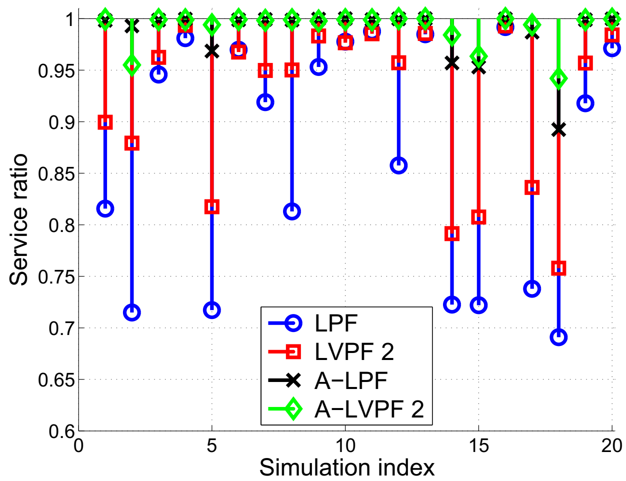


Figure 3: Service ratio with anxious scheduler, $K = 20$

[4] R. Agrawal, R. Berry, Jianwei Huang, V. Subramanian, *Optimal Scheduling for OFDMA Systems Signals*, Fortieth Asilomar Conference on Systems and Computers, 2006, pp. 1347-1351, Oct-Nov, 2006.

[5] J. Huang, V. G. Subramanian, R. Agrawal, R. Berry, *Joint scheduling and resource allocation in uplink OFDM systems for broadband wireless access networks*, Selected Areas in Communications, IEEE Journal on, vol. 27, issue 2, pp. 226-234, February 2009.

[6] Q. Bai, N. Passas, J. A. Nossek, *Scheduling and resource allocation in OFDM and FBMC systems: An interactive approach and performance comparison*, 2010 European Wireless Conference, April 2010.

[7] IEEE Std 802.16e, *IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems. Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1*, Feb. 2006.

[8] Q. Bai and J. A. Nossek, *On Two Cross-layer Assisted Resource Allocation Schemes in Multiuser Multicarrier Systems*, Wireless Communication Systems, 6th International Symposium on, Sept. 2009, pp. 186-190.

[9] B. Zerlin, M. T. Ivrlač, W. Utschick, J. A. Nossek, I. Viering and A. Klein, *Joint optimization of radio parameters in HSDPA*, in IEEE 61st Vehicular Technology Conference VTC 2005-Spring, vol. 1, pp. 295-299.

[10] R. G. Gallager, *Information Theory and Reliable Communication*, John Wiley and Son, 1968.

[11] Q. Bai and J. A. Nossek, *On Energy Efficient Cross-layer Assisted Resource Allocation in Multiuser Multicarrier Systems*, IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2009), Sept. 2009.