Physical Communication 8 (2013) 81–90

Contents lists available at ScienceDirect

Physical Communication

journal homepage: www.elsevier.com/locate/phycom

Full length article

Energy efficient QoS constrained scheduler for SC-FDMA uplink☆

Dan J. Dechene [a], Abdallah Shami [b,*]

[a] IBM Semiconductor Research and Development Center, Hopewell Junction, NY, United States

[b] Department of Electrical and Computer Engineering, Western University, London, Ontario, Canada

ARTICLE INFO

Article history:
Received 30 August 2012
Accepted 2 September 2012
Available online 7 September 2012

Keywords:
SC-FDMA
Scheduling
Multiuser
QoS
QSI

ABSTRACT

In this paper we propose a framework for an energy efficient scheduler for multiuser SC-FDMA with queue state information (QSI) and quality of service (QoS) constraints. Resource allocation is formulated as a two-stage problem where resources are allocated in both time and frequency. The scheduling policy is obtained in two stages for the intra- and inter-user allocations respectively. A near optimal iterative allocation method is used for the inter-user allocation and the intra-user allocation policy is obtained using a constrained Markov decision process framework. Results are presented for the energy performance.

1. Introduction

Energy efficient communication is a long-standing issue in modern wireless communication systems. The mobile device radio accounts for a large portion of a mobile device's battery life, and as such, efficient use of radio resources can dramatically improve mobile device energy consumption. With the increased proliferation of smaller and faster devices, it has become essential to efficiently utilize mobile battery resources.

General radio resource allocation problems, particularly for systems such as orthogonal frequency division multiplexing (OFDM) fall into two major classifications, namely the rate and margin adaption (RA and MA) problems [1]. RA problems try to allocate resource to maximize system throughput for a given power constraint, while MA problems try to minimize transmission power while maintaining a minimum throughput guarantee. The latter is used for energy efficient scheduling.

In recent years, MA problems have been well-studied for a general OFDMA transmission system [1,2]. However, more modern systems, such as 3GPP-LTE, utilize localized single carrier frequency division multiple access (SC-FDMA) at the physical layer for uplink transmissions. This is due to the improved peak to average power ratio (PAPR) when employing SC-FDMA. Unfortunately, contiguous frequency block assignment in SC-FDMA eliminates direct application of the previous MA framework described above. Furthermore, the finite set of modulation and coding schemes (MCSs) dramatically increases the optimal allocation complexity.

In this paper we propose a cross-layer resource allocation scheme which minimizes the weighted average applied power per user while ensuring quality of service (QoS) requirements are met. The contributions are presented in two parts. In the first part, the dynamic scheduling policy framework originally presented in [3] is used to allocate user data in order to minimize the overall average power expenditure for intra-user allocation while meeting long-term QoS constraints. Secondly, we propose a near optimal, low complexity online iterative MA allocation scheme for SC-FDMA to transmit the intra-user allocated data. This inter-user stage minimizes the overall applied power per subframe required to transmit the user data online subject to channel conditions.

The remainder of this paper is divided as follows. In Section 2 we overview the details of the employed

☆ Invited paper.

* Corresponding author.
E-mail addresses: ddechene@ieee.org (D.J. Dechene), ashami@eng.uwo.ca (A. Shami).

1874-4907/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.phycom.2012.09.001

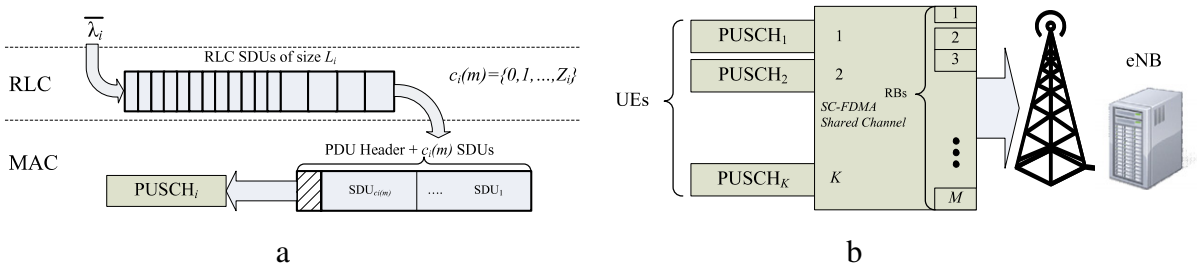


Fig. 1. System model: (a) each UE, (b) overall system.

uplink system model including the channel and scheduling models and in Section 3 we describe the scheduling ideology. In Section 4 simulation results are provided while in Section 5, conclusions are drawn on this work.

2. System model

The system model is shown in Fig. 1. We assume that there are K users (denoted as UEs) within a single cell, communicating with a single base station (denoted as an eNB). Since we are concerned with resource allocation within a single cell, for the purpose of this paper, it is assumed that intercell interference is negligible. The cell spectrum is divided into N_{sub} subcarriers which are grouped into M resource blocks. Each resource block (RB) is comprised of 12 equivalent subcarriers. Without loss of generality we assume there is an integer number (M) of RBs available for allocation. The system is assumed to be operating in FDD mode.

There are N_{sym} symbols per subcarrier in a given subframe where the exact number of subcarriers depends on the uplink configuration. The physical uplink shared channel (PUSCH) is used for transmission of uplink data and comprises a portion of symbols along with other control channels. For the purpose of this paper, it is assumed that the PUSCH occupies N_{sym}–N_{ctrl} symbols per subcarrier, per subframe where N_{ctrl} is the number of symbols used for all other physical channels and signalling.

The time scheduling horizon is divided into small subframes consisting of two LTE time slots and has a duration 1 ms. An example layout of the time scheduling horizon is shown in Fig. 2. During each subframe, where m is to denote the mth subframe, users can transmit up to $T_i(m)$ bits of data as determined by the eNB. The long-term average service rate experienced by a user is μ_i SDUs per second. Each UE also has a power-allocation priority weight α_i which can be used to denote the relative importance of user in terms of minimizing their individual power consumption.

Each UE receives all uplink traffic from upper layers of their protocol stack destined for transmission to the eNB. Each UE's traffic has associated QoS parameters $\{D_i, L_i, \bar{\lambda}_i, B_i, P_{\text{drop},i}\}$ which denotes the maximum tolerable average delay, service data unit (SDU) length, average arrival rate, buffer size at the radio link control (RLC) layer and maximum buffer dropping rate of SDU respectively for that user. Each stream may represent a broad service class (such as voice over IP or video) or a particular application-layer stream being used at the time. Each incoming stream is stored in a finite-length first-in, first-out (FIFO) buffer where incoming SDUs are dropped when the buffer is full. There is one traffic stream for each UE. For clarity for the reader, Table 1 summarizes the frequently used notation in this paper.

Table 1
Frequently used notation.

Quantity	Symbol
Number of UEs	K
Number of RBs in frequency	M
UE index	i
Subchannel index	k
Subframe number	m
Target block error rate	BLER_{tgt}
Set of allocated subchannels to UE i	$\mathcal{N}_i$
Transport block header size	L_{hdr}
Transport block size in subframe m	$T_i(m)$
Average delay target for UE i	D_i
SDU length for UE i	L_i
Average SDU arrival rate for UE i	$\bar{\lambda}_i$
SDU buffer size for UE i	B_i
Target SDU dropping rate for UE i	$P_{\text{drop},i}$
Maximum number of transmitted SDUs per UE i in a subframe	Z_i
Allocated SDUs in subframe m for UE i	$c_i(m)$
SDU rate state-space for UE i	$\mathcal{C}_i$
Buffer level of UE i at the beginning of subframe m	$u_i(m)$
Arrivals to buffer of UE i during subframe m	$A_i(m)$
Scheduling policy	Ω
Steady-state policy distribution function for UE i	$\theta_i(c_i, u_i\|\Omega)$
Steady-state action probability	$\pi_i(c_i\|\Omega)$
Reference SNR	γ_0
Average SNR in subchannel k for UE i	$\gamma_{i,k}$
Effective SNR for a set of subchannels for UE i	$\gamma_{i,\text{eff}}$

2.1. Assumptions

The following assumptions are made for the remainder of this paper.

- The CSI matrix corresponding to the channel between each UE and the eNB over all RBs is available at the eNB error free.
- The eNB feedback channel informs UEs in advance of the resource blocks and quantity of data for transmission for a user during any uplink subframe.
- The eNB has knowledge about buffer occupancy levels and QoS parameters of each UE.

Similar assumptions regarding the CSI and feedback channel are made in many related cross-layer works including [4,5].

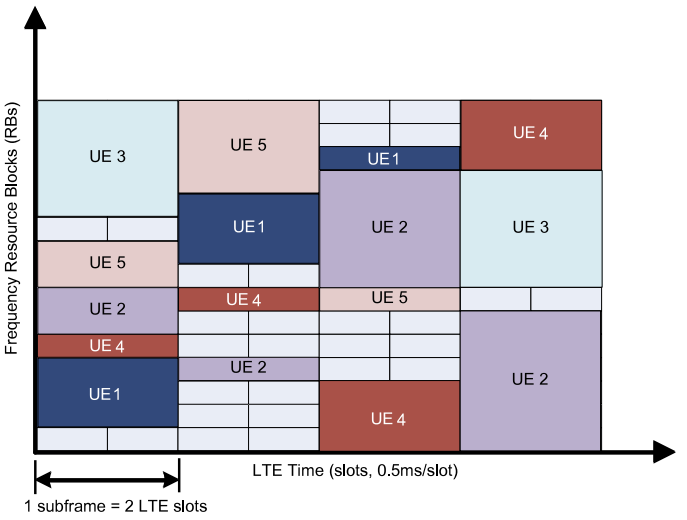


Fig. 2. Time scheduling horizon.

2.2. Finite transport block sizes

In most packet based transmission systems, due to physical limitations, users can only service a finite number of SDUs from the queue during a small period of time.[1] Assuming there is no SDU segmentation at the physical layer before transmission, we can denote Z_i as the maximum number of SDUs that can be serviced during any time subframe m by UE i where $c_i \in \mathcal{C}_i = \{0, 1, \ldots, Z_i\}$ and c_i is the number of SDUs serviced during any subframe. As a result, the eligible sizes of each UE's transport block are also a finite set given as $\mathcal{T}_i = \{0, L_i + L_{\text{hdr}}, 2L_i + L_{\text{hdr}}, \ldots, Z_iL_i + L_{\text{hdr}}\}$ where L_i is the SDU length in bits and L_{hdr} is the header size.[2] During each subframe, the eNB allocates to user i an uplink slot of $T_i(m) \in \mathcal{T}_i$ bits.

2.3. Channel state information (CSI)

Channel state information is assumed available at the eNB for the next subframe. We assume this information is available error free. The channel is modelled as block fading where the channel is static for the duration of a subframe and independent from subframe to subframe. The channel experienced from UE to UE is assumed independent. For now, the channel transfer function for each RB is also assumed to be independent of adjacent RBs and each channel follows the Rayleigh SNR distribution given as

$$p(\gamma) = \frac{1}{\gamma_0} \exp\left(-\frac{\gamma}{\gamma_0}\right) \tag{1}$$

where $\gamma_{i,k}(m)$ will be used to denote the uplink channel of user i over RB k in subframe m.

[1] Such a rationale was previously described in [3].

[2] No header is required if a UE user does not transmit any SDUs during a given subframe.

2.4. Queue evolution

From slot m to slot $m+1$ the evolution of the RLC queue of each user evolves according to

$$u_i(m+1) = \min\{B_i, \max\{0, u_i(m) - c_i(m)\} + A_i(m)\} \tag{2}$$

where $u_i(m)$ is the number of SDUs in queue i at the beginning of subframe m, $A_i(m)$ is the number of SDUs arriving during subframe m to the queue and $c_i(m)$ is the number of SDUs taken from queue i during subframe m.

3. Resource allocation framework

The proposed resource allocation algorithm is divided into intra-user and inter-user allocation stages. The intra-user stage controls the number of SDUs allocated per subframe to meet the individual loss, delay and throughput requirements of all each users' stream while minimizing the average weighted energy expenditure, while the inter-user stage allocates channel resources to individual UEs to minimize the per subframe weighted power allocation subject to the conditions of the wireless channel.

3.1. Intra-user allocation

The intra-user allocation operates as follows and is formulated as a constrained Markov decision process (MDP). The intra-user allocation chooses the number of SDUs $\{c_i(m)|0 \le c_i(m) \le Z_i\}$ for each UE i during subframe m and where Z_i denotes the maximum number of SDUs that can be transmitted by UE i during any subframe. The system state of UE i is denoted by its buffer level and the action space of the constrained MDP describes the number of SDUs that can be transmitted during a subframe (or set of values for c_i) subject to the computed randomized policy. The set of all feasible action spaces across all UEs is $\mathcal{C}$ (known a priori) and as described in Section 2.2.

Let $\theta_i(c_i, u_i|\mathbf{\Omega})$ be a steady-state distribution function that exists for a particular policy $\mathbf{\Omega}$ where u_i is the buffer

occupancy level of UE i. The solution to the allocation problem for any scheduling policy $\mathbf{\Omega}$ is a random policy described by this distribution function $\theta_i(c_i, u_i|\mathbf{\Omega})$ which denotes the probability of choosing the action c_i as the number SDUs for transmission given that UE i is in state u_i. The aforementioned policy is derived for all UEs in a similar fashion to [3]. The goal of the optimization formulation for intra-user scheduling is to find $\theta_i(c_i, u_i|\mathbf{\Omega})$ for all c_i, u_i, as well as i that minimizes the average applied transmission power. The resultant policy is coupled by $\mathbf{\Omega}$ which defines the scheduling actions for each queue i and each queue state $u_i \in \mathcal{U}_i$. Application of this policy in each subframe m determines the number $T_i(m)$ in bits to be transmitted for UE i (where $T_i(m) = L_i c_i(m) + L_{\text{hdr}}$ for $c_i(m) > 0$ or $T_i(m) = 0$ when $c_i(m) = 0$), $c_i(m)$ is c_i chosen randomly at time m with probability defined by $\theta_i(c_i, u_i|\mathbf{\Omega})$ and L_{hdr} is the size of the subframe header. $T_i(m)$ for all i is then allocated subject to the inter-user allocation algorithm. As the cost function of the constrained MDP problem relies on knowledge about the inter-user allocation algorithm, the estimated average applied power per action is computed as discussed in Section 3.3.

The intra-user allocation constraints are on throughput and delay. These are measured as follows.

3.1.1. Station throughput

Throughput is measured as the amount of goodput over the channel. Assuming each transmission experiences a block error rate of BLER_{tgt} the average throughput is given by[3]

$$\text{Throughput} = \mathbb{E}_m[T_i(m)](1 - \text{BLER}_{\text{tgt}}). \tag{3}$$

Further, we note that dropping probability of a given queue is related to the service rate as

$$P_{\text{drop},i} = 1 - \frac{\mathbb{E}_m[T_i(m)]}{\overline{\lambda}_i}. \tag{4}$$

3.1.2. SDU delay

The SDU delay can be found from Little's Theorem. Here, the average queueing delay can be given as

$$\mathcal{D}_i = \frac{\overline{q}_i}{\lambda_{q,i} T_f} \tag{5}$$

where $\overline{q}_i$ is the average queue size and $\lambda_{q,i}$ is the average enqueued arrival rate for queue i. By design we can express $\overline{q}_i$ using the steady-state distribution $\theta_i(c_i, u_i|\mathbf{\Omega})$ as:

$$\overline{q}_i = \sum_{u_i \in \mathcal{U}_i} u_i \sum_{c_i \in \mathcal{C}_i} \theta_i(c_i, u_i|\mathbf{\Omega}) \tag{6}$$

and since $\lambda_{q,i}$ is also equal to the average service rate in steady-state, it can be expressed as

$$\lambda_{q,i} = \sum_{u_i \in \mathcal{U}_i} \sum_{c_i \in \mathcal{C}_i} \min(c_i, u_i) \theta_i(c_i, u_i|\mathbf{\Omega}). \tag{7}$$

We also note that $\lambda_{q,i}$ is related to the throughput as

$$\mathbb{E}_m[T_i(m)] = \lambda_{q,i} L_i. \tag{8}$$

[3] We note in the above equation, the number of SDUs does not appear. This is important to note as in general SDUs encoded at the physical layer are sent as one block. If the block is erroneous, all SDUs within the subframe are erroneous.

3.1.3. Per-queue transition probability

The transition of each queue is solely based on the arrivals and departures from that queue. For Poisson arrivals with average rate $\overline{\lambda}_i$ in SDUs per subframe, the transition probability is given as

$$p^{c_i}_{u_i;u_i'} = \begin{cases} \Pr\left[A_i(m) = u_i' - [u_i - \min(u_i, c_i)]\right], & u_i' < B_i \\ \sum_{j=B_i-[u_i-\min(u_i,c_i)]}^{\infty} \Pr[A_i(m) = j], & u_i' = B_i \end{cases} \tag{9}$$

where

$$\Pr[A_i(m) = k] = \begin{cases} \dfrac{\overline{\lambda_i}^k \exp(-\overline{\lambda_i})}{k!}, & \text{if } k \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

for a given average arrival rate $\overline{\lambda_i}$, buffer size B_i and all eligible SDU service rates c_i.

3.1.4. Per user objective function

The transmission cost is found for each $c \in \mathcal{C}$. Let $\mathcal{P}(c)$ be the joint cost of choosing action c (action $c_1, c_2, \ldots, c_K$ for each UE). As with [3], the marginal cost for each queue can be obtained using $\mathcal{P}(c)$ combined with the steady state distribution as in Eq. (23) of [3] to obtain the marginal cost function used in the intra-user constrained optimization.

$\mathcal{P}(c)$ is dependent on the channel. Moreover, it is difficult to obtain a closed form expression on $\mathcal{P}(c)$ for all c, particularly since this is dependent on the average performance of the inter-user allocation stage. The method used to obtain $\mathcal{P}(c)$ is discussed in Section 3.3. Once this is determined, the cost function is obtained as follows.

First, the average marginal cost for taking an action $c_1 = x$ in user 1 for example can be given as

$$\Upsilon_{1,x} = \sum_{c_2 \in \mathcal{C}_2} \cdots \sum_{c_K \in \mathcal{C}_K} P(x, c_2, \ldots, c_K) \cdot \pi_2(c_2|\mathbf{\Omega}) \times \cdots \times \pi_K(c_K|\mathbf{\Omega}) \tag{11}$$

where there are $i-1$ summations. Similar expressions can be found for all actions $c_i \in \mathcal{C}_i$ and found for all users $k = 1, \ldots, K$ and where

$$\pi_i(x|\mathbf{\Omega}) = \sum_{u_i \in \mathcal{U}_i} \theta(x, u_i|\mathbf{\Omega}), \quad x \in \mathcal{C}_i \tag{12}$$

$P(c_1, c_2, \ldots, c_K)$ denotes the average weighted power allocated to transmit $\{c_1, c_2, \ldots, c_K\}$ SDUs from each UE i. In compact notation we denote this $\mathcal{P}(c)$ where each $c \in \mathcal{C}$ corresponds to a set $\{c_1, c_2, \ldots, c_K\}$ for all users.

By design, the steady-state distribution $\theta_i(c_i, u_i|\mathbf{\Omega})$ must also satisfy the following balance property

$$\sum_{u_i' \in \mathcal{U}_i} \sum_{c_i' \in \mathcal{C}_i} \theta(c_i', u_i'|\mathbf{\Omega}) p^{c_i'}_{u_i';u_i} = \sum_{c_i \in \mathcal{C}_i} \theta(c_i, u_i|\mathbf{\Omega}), \quad \forall u_i. \tag{13}$$

3.1.5. Iterative policy solver

The above steady-state action probabilities are coupled through the policy $\mathbf{\Omega}$. The value $P(c)$ is the total power

associated with taking actions c_1 through c_K in each UE (or one for each state $c \in \mathcal{C}$) found earlier. Here we need to highlight that the above expression contains the steady-state probability of choosing an action for each user, the result of which implies that it is not possible to directly decouple and consider each user independently. As a result, we employ the per-user iterative policy solver developed in [3]. The solver operates as follows.

Firstly, each user policy vector is initialized to

$$\pi_i(x|\mathbf{\Omega}^{(0)}) = \frac{1}{Z_i + 1}, \quad \forall x, i \tag{14}$$

where $\mathbf{\Omega}^{(n)}$ denotes the policy computed at iteration n. Let $\mathcal{K}$ denote the set of users where $\mathcal{K} = \{1, 2, \ldots, K\}$ and n denote the iteration number where initially $n = 1$. At each iteration, $i^* = (n \bmod K) + 1$ where n is incremented for each iteration. For each iteration we solve for $\pi_{i^*}(x|\mathbf{\Omega}^{(n)})$ as follows.

As in [3], the constrained MDP problem at each iteration is solved using convex linear programming (LP) techniques formulated as $\arg\min_{\mathbf{x}} \mathbf{c}^T\mathbf{x}$, subject to $\mathbf{A}\mathbf{x} \leq \mathbf{b}$, $\mathbf{A}_{\text{eq}}\mathbf{x} = \mathbf{b}_{\text{eq}}$, $\mathbf{x} \geq 0$ where $\mathbf{A}$ and $\mathbf{A}_{\text{eq}}$ are matrices and $\mathbf{x}$, $\mathbf{b}$, $\mathbf{b}_{\text{eq}}$ and $\mathbf{c}$ are column vectors. The vector $\mathbf{x}$ is the solution to the optimization problem. In our problem, the elements are given as

$$\mathbf{x} = [\boldsymbol{\theta}_{i^*}(\mathcal{C}_{i^*}, 0|\mathbf{\Omega}^{(n)}), \ldots, \boldsymbol{\theta}_{i^*}(\mathcal{C}_{i^*}, B_{i^*}|\mathbf{\Omega}^{(n)})]^T \tag{15}$$

with each $\boldsymbol{\theta}_{i^*}(\mathcal{C}_{i^*}, u_{i^*}|\mathbf{\Omega}^{(n)})$ being a row vector with entries for each $c_{i^*} = 0, 1, \ldots, Z_{i^*}$.

The objective function is of the form $\mathbf{c}^T\mathbf{x}$. The vector $\mathbf{c}$ is comprised of the total power cost for taking an action. Each entry of $\mathbf{c}$ corresponds to the entry in $\mathbf{x}$ with the value of entries in $\mathbf{c}$ given by $\Upsilon_{i^*,c_{i^*}}$ in (11).

$$\mathbf{c} = [\underbrace{\Upsilon_{i^*,1}, \ldots, \Upsilon_{i^*,Z_{i^*}+1}}_{1}, \overbrace{\cdots}^{2\cdots B_{i^*}}, \underbrace{\Upsilon_{i^*,1}, \ldots, \Upsilon_{i^*,Z_{i^*}+1}}_{B_{i^*}+1}]. \tag{16}$$

The equality constraints are comprised of the balance equations and the causality constraint (total probability space). In matrix form, the balance equations can be expressed as $\mathbf{P} \times \mathbf{x} = \mathbf{\Phi}_0 \times \mathbf{x}$ where $\mathbf{P}$ is given by

$$\mathbf{P} = \begin{bmatrix} \mathbf{p}_{0;0}^{\mathcal{C}_{i^*}} & \cdots & \cdots & \mathbf{p}_{B_{i^*};0}^{\mathcal{C}_{i^*}} \\ \vdots & \mathbf{p}_{1;1}^{\mathcal{C}_{i^*}} & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{p}_{0;B_{i^*}}^{\mathcal{C}_{i^*}} & \cdots & \cdots & \mathbf{p}_{B_{i^*};B_{i^*}}^{\mathcal{C}_{i^*}} \end{bmatrix} \tag{17}$$

with $\mathbf{p}_{q;q'}^{\mathcal{C}_{i^*}}$ as a $1 \times (Z_{i^*} + 1)$ row vector with entries

$$\mathbf{p}_{q;q'}^{\mathcal{C}_{i^*}} = [p_{q;q'}^{1}, \ldots, p_{q;q'}^{Z_{i^*}+1}] \tag{18}$$

and the quantity $\mathbf{\Phi}_0$ is given as the $B_{i^*} + 1$ row matrix

$$\mathbf{\Phi}_0 = \begin{bmatrix} \mathbf{1}_{1\times(Z_{i^*}+1)} & 0 & \cdots & 0 \\ 0 & \mathbf{1}_{1\times(Z_{i^*}+1)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{1}_{1\times(Z_{i^*}+1)} \end{bmatrix}. \tag{19}$$

Combining the above with the causality constraint on the total probability space we have our overall equality constraints given as

$$\mathbf{A}_{\text{eq}} = \begin{bmatrix} \mathbf{P} - \mathbf{\Phi}_0 \\ \mathbf{1}_{1\times((Z_{i^*}+1)(B_{i^*}+1))} \end{bmatrix} \tag{20}$$

$$\mathbf{b}_{\text{eq}} = \begin{bmatrix} \mathbf{0}_{1\times(B_{i^*}+1)} & 1 \end{bmatrix}^T. \tag{21}$$

The inequality constraints are used to describe the throughput and delay constraints. These constraints are given in two parts as

$$\mathbf{A} = \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} \tag{22}$$

where $\mathbf{w}_1$ is given as

$$\mathbf{w}_1 = -[\boldsymbol{\chi}_{i^*:n}(\mathcal{C}_{i^*}, 0), \ldots, \boldsymbol{\chi}_{i^*:n}(\mathcal{C}_{i^*}, B_{i^*})] \tag{23}$$

where $\boldsymbol{\chi}_{i^*:n}(\mathcal{C}_{i^*}, u_{i^*})$ is a row vector with entries $\chi_{i^*:n}(c_{i^*}, u_{i^*})$ for all $c_{i^*} \in \mathcal{C}_{i^*}$ and $\mathbf{z}_1$ is given as

$$\mathbf{z}_1 = -\bar{\lambda}_{i^*}(1 - P_{\text{drop},i^*})T_f. \tag{24}$$

Finally, $\mathbf{w}_2$ is given as

$$\mathbf{w}_2 = \mathbf{Q} \times \mathbf{\Phi}_0 - \mathcal{D}_{i^*}\mathbf{U} \qquad \mathbf{z}_2 = 0 \tag{25}$$

where $\mathbf{Q} = [0, 1, \ldots, B_{i^*}]$ and $\mathbf{U}$ is given in (26).

$$\mathbf{U} = [\min(0, 0), \min(1, 0), \ldots, \min(Z_{i^*}, 0), \min(0, 1), \ldots, \min(Z_{i^*}, B_{i^*})] = -\mathbf{w}_1. \tag{26}$$

The total average weighted power in the system at iteration n is given as

$$\mathcal{P}^{(n)} = \sum_{c_1 \in \mathcal{C}_1} \cdots \sum_{c_K \in \mathcal{C}_K} P(c_1, c_2, \ldots, c_K) \cdot \pi_1(c_1|\mathbf{\Omega}^{(n)}) \times \cdots \times \pi_K(c_K|\mathbf{\Omega}^{(n)}). \tag{27}$$

The process continues iteratively until reaching one of the following stopping conditions

(i) $|\mathcal{P}^{(n-k)} - \mathcal{P}^{(n-k-1)}| < \epsilon$, $k = 0, \ldots, K-1$, $n \geq K$

(ii) $n > \text{MAX}_{\text{iter}}$

where MAX_{iter} is the preset maximum number of iterations and ϵ is a small positive number.

3.2. Inter-user allocation

The inter-user allocation occurs as follows. During each subframe, each of the K users transmits a single transport block of $T_i(m)$ bits. It is assumed that eligible transport block sizes are an integer number of SDUs plus a header (i.e., no SDU segmentation is required by the RLC, c_i are integer values as discussed above).

We previously noted the uplink physical layer employs SC-FDMA. Resources during any given subframe must therefore be allocated contiguously in frequency, and only a single contiguous transport block can be allocated per subframe.

This allocation is done employing an iterative, near-optimal allocation technique using Algorithm 1. This algorithm iteratively allocates resources to users to maximize

Algorithm 1 Iterative Power Efficient Resource Allocation

$\mathcal{N} = \{1, 2, \ldots, M\}$
$\mathcal{N}_i = \emptyset, \quad \forall i \in \mathcal{K}$
$\mathcal{K}^{(a)} = \mathcal{K}$
$\mathcal{N}_i^{(f)} = \mathcal{N}, \quad \forall i \in \mathcal{K}$
while $|\mathcal{N}| > |\mathcal{K}^{(a)}|$ **do**
  for $i \in \mathcal{K}$ **do**
    if $\mathcal{N}_i \neq \emptyset$ **then**
      for $j \in \mathcal{N}_i^{(f)} \cap \mathcal{N}$ **do**
        $\Delta p_{i,j} = \alpha_i(P(\mathcal{N}_i, T_i, \boldsymbol{\gamma}) - P(\mathcal{N}_i \cup j, T_i, \boldsymbol{\gamma}))$
      end for
    else
      for $\mathcal{N}$ **do**
        $p_{i,j} = \alpha_i P(\mathcal{N}_i, T_i, \boldsymbol{\gamma})$
      end for
      $\Delta p_{i,j} = \min(\{p_{i,j}, j \in \mathcal{N} \setminus \arg\min_{j^* \in \mathcal{N}}(p_{i,j^*})\}) - \min(\{p_{i,j}, \forall j \in \mathcal{N}\})$
    end if
  end for
  if $\max(\Delta p_{i,j}) < 0$ **then**
    break
  end if
  $(i^*, j^*) = \arg\max_{i,j} \Delta p_{i,j}$
  $\mathcal{K}^{(a)} = \mathcal{K}^{(a)} \setminus i^*$
  $\mathcal{N}_{i^*} = \mathcal{N}_{i^*} \cup j^*$
  $\mathcal{N}_{i^*}^{(f)} = \{\min(\mathcal{N}_{i^*}) - 1, \max(\mathcal{N}_{i^*}) + 1\} \cap \mathcal{N}$
  $\mathcal{N} = \mathcal{N} \setminus j^*$
end while
if $|\mathcal{K}^{(a)}| \neq \emptyset$ **then**
  for $i \in \mathcal{K}^{(a)}$ **do**
    for $j \in \mathcal{N}$ **do**
      $p_{i,j} = \alpha_i P(\mathcal{N}_i, T_i, \boldsymbol{\gamma})$
    end for
  end for
  while $|\mathcal{K}^{(a)}| \neq \emptyset$ **do**
    $(i^*, j^*) = \arg\min_{i \in \mathcal{K}^{(a)}, j \in \mathcal{N}} p_{i,j}$
    $\mathcal{K}^{(a)} = \mathcal{K}^{(a)} \setminus i^*$
    $\mathcal{N}_{i^*} = j^*$
    $\mathcal{N} = \mathcal{N} \setminus j^*$
  end while
end if

the power level gain at each iteration similar to the block MA allocation in [6], which was based on the RA allocation algorithm in [7]. In the Appendix we show that this method performs near-optimal MA subcarrier and power allocation when the number of users scheduled per subframe is less than half the number of the RBs (i.e., $2K \leq M$). For now, we assume that this is the case. For a given set of intra-user allocations $\{T_i(m), \forall i\}$, the algorithm allocates power and RBs to each user. The selected power level is given as the ratio of the required SNR to the measured SNR for a given target error rate. We utilize the block outage probability from [8] to model the block error rate (BLER) of coded transmissions.[4] This is a function of the number of resource blocks (total number of symbols) and the data rate. Given a target BLER, a data rate T_i (measured in bits per subframe) and a set of resource blocks $\mathcal{N}_i$, we know that

$$\text{BLER}(\gamma_{i,\text{eff}}^{(r)}, \mathcal{N}_i, T_i) \approx Q\left(\frac{\log(1+\gamma_{i,\text{eff}}^{(r)}) - \frac{\log(2)T_i}{\eta(\mathcal{N}_i)}}{\sqrt{\frac{2}{\eta(\mathcal{N}_i)}\frac{\gamma_{i,\text{eff}}^{(r)}}{1+\gamma_{i,\text{eff}}^{(r)}}}}\right) \tag{28}$$

where $\gamma_{i,\text{eff}}^{(r)}$ is the SNR level required for a given BLER.

In order to use the above for margin adaptive resource allocation, we must solve for $\gamma_{i,\text{eff}}^{(r)}$ as a function of T_i and $\mathcal{N}_i$. From this, one can obtain the required allocation power from the SNR gap between the measured effective SNR and the required SNR. Due to the monotonicity of the Q-function arguments, the above can be solved efficiently using bisection techniques. Alternatively, a more computationally efficient method is to obtain a least squares approximation to the above as a function of data rate, target BLER and the number of RBs allocated (similar to the approach in [4]). We found the following fitting function closely approximates the SNR as a function of data rate

$$\gamma_{i,\text{eff}}^{(r)} \approx a_x \exp(b_x T_i) - \gamma_{0,x} \tag{29}$$

where $x = |\mathcal{N}_i|$ and validations of this approximation is given in Section 4.1.

Using the above, the required applied power is given as

$$P(\mathcal{N}_i, T_i, \boldsymbol{\gamma}) = \frac{\gamma_{i,\text{eff}}^{(r)}}{\gamma_{i,\text{eff}}^{(m)}} = \frac{a_{|\mathcal{N}_i|}\exp(b_{|\mathcal{N}_i|}T_i) - \gamma_{0,|\mathcal{N}_i|}}{\frac{1}{|\mathcal{N}_i|^2}\sum_{k \in \mathcal{N}_i}\gamma_{i,k}} \tag{30}$$

where $\mathcal{N}_i$ is the set of RBs that we allocate to a UE and T_i is the number of bits for transmission. Parameters a_x, b_x, and $\gamma_{0,x}$ are given from the least-squares approximation derived as follows and depend on the target block error rate of the channel (BLER_{tgt}).

3.3. Determining power applied per state

Determination of $\mathcal{P}(c)$ for intra-user allocation in Section 3.1 requires knowledge of the expected power allocation of the inter-user allocation in Section 3.2.

For any state vector c, the bitrate T_i is easily obtained as $T_i = c_i L_i + L_{\text{hdr}}$ for $c_i > 0$ or $T_i = 0$ for $c = 0$ in state c (recalling c is the joint state of c_i for all i). From this one can obtain $T_i, \forall i$ for use in Algorithm 1 and find the weighted transmission power using (30) for a given channel realization. A measurement of the average weighted power of state c (i.e., $\mathcal{P}(c)$) can be obtained by averaging the weighted transmission power over all realizations of the channel matrix and over all possible system states c. Since we assume the channel is continuous, and the number of resource blocks is large, it is difficult to do this for all realizations, we can however

[4] While here we utilize BLOP to model the error rate of coded transmissions, extensions are trivial given measurements of BLER performance obtained via proper training and calibration or in cases where analytical expressions are obtainable.

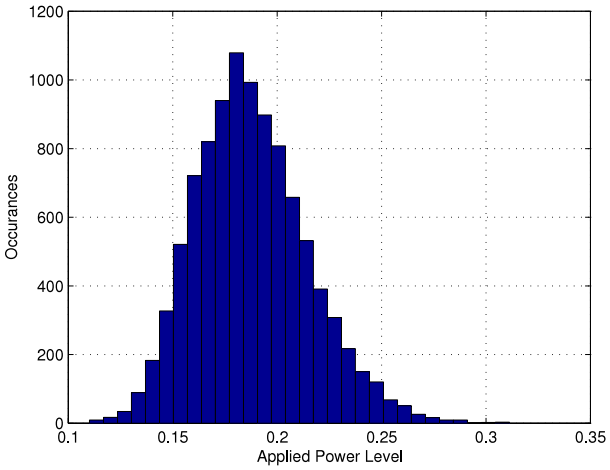


Fig. 3. Histogram of power applied per iteration.

obtain $\mathcal{P}(c)$ by averaging over a finite number of random realizations for each c.

Consider the following example where there are five UEs, and 24 RBs and each user is transmitting 400 bits (for this state c, i.e., $T_1 = T_2 = \cdots = T_5$) over one subframe. In Fig. 3, we see a histogram of the actual applied power per iteration (instantaneous $\widetilde{\mathcal{P}(c)}$ over 10 000 iterations) for this state c (where RBs and power are allocated at each iteration based on the channel CSI). What we observe is that this number of random realizations provides a smooth distribution of the average applied power implying a sufficient number of samples. For any c, the average applied power ($\mathcal{P}(c)$) is measured as the mean instantaneous value (i.e., 0.1879 or −7.26 dB in this example). Similar values can be obtained for all other $c \in \mathcal{C}$.

3.4. System complexity

In general all system parameters including the intra-user policy and average applied power per state can be measured and calculated in advance and stored in memory. The proposed framework is functional for a small to moderate number of instantaneous users as the size of $\mathcal{C}$ is given as

$$|\mathcal{C}| = \prod_{i=1}^{K} (Z_i + 1). \tag{31}$$

For the case above, where we assume $Z_i = 4$, we see that $|\mathcal{C}| = 5^{(4+1)} = 3125$. While we observe that the number of states increases exponentially, it is still considerably more efficient than consideration of the joint buffer occupancy states. For example, if each user had a buffer size of up to 25 SDUs, the resulting space would be $5^{(25+1)} = 1.49 \times 10^{18}$.

4. Results

Results are presented with universal parameters summarized in Table 2. Details are provided for the least-squares approximation as well as the applied power versus various parameters.

Table 2
Simulation parameters.

Parameter	Value
γ_0	10 dB
N_{sym}	14
N_{ctrl}	3
BLER_{tgt}	10%
Number of subframes	10 000
T_f	1 ms
Number of RBs (M)	24
Number of UEs (K)	2
ϵ	10^{-7}
L_{hdr}	30 bits
α_i	1 $\forall i$
$P_{\text{drop},i}$	0.1% $\forall i$
B_i	25 SDUs $\forall i$
$\bar{\lambda}_i$	1.5 SDUs per subframe, $\forall i$
D_i	4 subframes $\forall i$
Z_i	4$\forall i$
T_i	$150 + 50i$, $\forall i$
MAX_{iter}	10 K

4.1. Least-squares applied power approximation

The justification behind use of the pre-described fit function is shown in Fig. 4. Here we see in this comparison two, four, and eight RBs with $\text{BLER}_{\text{tgt}} = 10\%$. The least-squares approximation is shown to hold tightly to the actual BLER function; providing for a more tractable computation of the required SNR level and justifying its use as a suitable alternative in computation of the required SNR.

4.2. Average applied power

Results for the average applied power are shown in Fig. 5. From these we observe several trends. Firstly, we observe a very low impact on the delay beyond two subframes in terms of percentage difference. This finding is consistent with the results found in [3]. The impact on SDU size is significantly larger in terms of average power applied. We also observe the impact on the number of users in Fig. 5(b). Here as expected the system expends additional power to accommodate the increase in users.

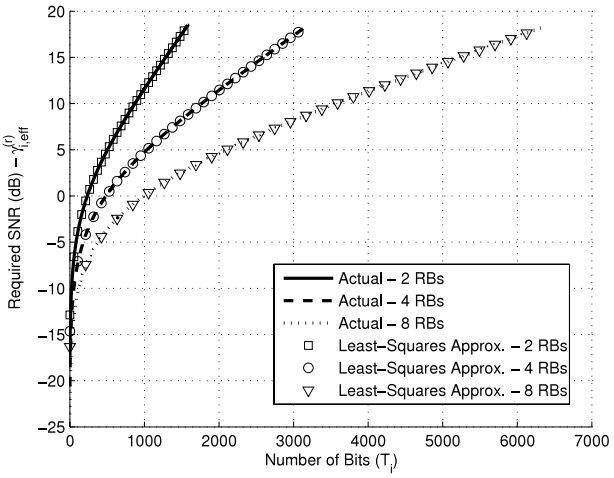
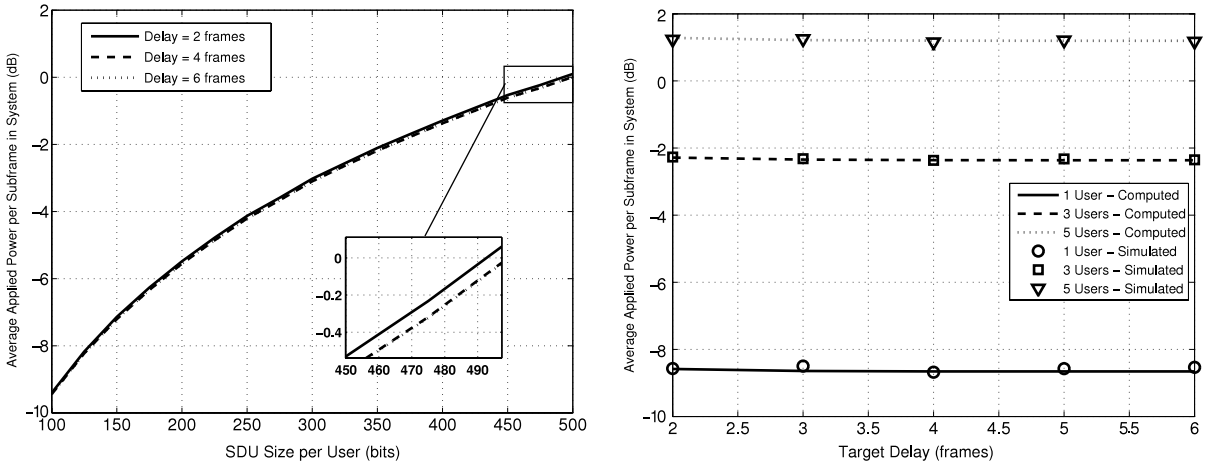


Fig. 4. Least-squares approximation accuracy.



(a) Versus target average delay and SDU size.

(b) Versus target average delay and number of UEs.

Fig. 5. Average applied power per subframe versus system parameters.

In this graph the computed and simulated results are shown (computed is given as the solution to the optimal policy, simulated obtained through implementation of the computed policy). The simulated results closely follow the expected scheduling policy performance.

5. Conclusion

In this paper we designed and evaluated the design of an energy efficient scheduler for multiuser SC-FDMA uplink. By exploiting part of our previously designed iterative scheduling technique, with a near optimal iterative resource allocation mechanism, a low-complexity scheduling policy was obtained. The proposed design was compared versus various parameters.

Our future work will look to extend these results in several ways. Firstly, after examining Fig. 3 we observe that the applied power per state appears at a glance to follow certain well-known distribution functions. In order to reduce the dependency on obtaining an estimate of the applied average power as discussed in Section 3.3, we look to better characterize the inter-user channel. In this way, it may be possible to obtain an exact or approximate analytical expression for the average expected power to eliminate this step. We note however that this is highly dependent on the underlying channel model. The second extension will focus on combining multiple classes of traffic into a single UE at the intra-user allocation stage. In this way, UEs will multiplex multiple classes of QoS constrained traffic (such as a simultaneous voice and data stream) over the shared uplink channel while still minimizing energy expenditure.

Appendix. Optimal gap

In order to address the performance of the iterative allocation algorithm described in Algorithm 1, we compare its performance with the optimal allocation. The optimal optimizations formulation is a traditional margin adaptive power and subcarrier allocation with subcarrier contiguity constraints in frequency and is formulated as follows.

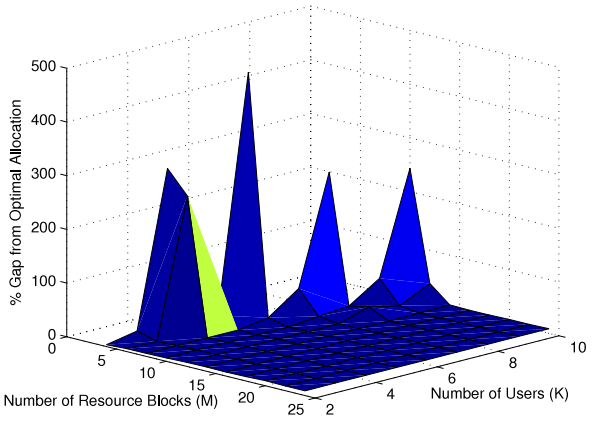


Fig. 6. Optimal allocation gap.

The formulation can be done using a similar approach as that used in [7], however modified to operate for the MA resource allocation problem. In this fashion, the contiguity constraints are exploited in a manner that reduces the binary search space.

The optimization problem is solved at each subframe m. For brevity of notation, the index m is dropped however all quantities are assumed to be specific to subframe m. The problem can be expressed as a general set-packing problem and formulated using binary programming as

$$\min_{\mathbf{x}} \ \mathbf{c}^T\mathbf{x} \tag{32}$$

$$\text{s.t. } \mathbf{A}\mathbf{x} \leq \mathbf{1}_M, \qquad \mathbf{A}_{\text{eq}}\mathbf{x} = \mathbf{1}_K, \quad x_j \in \{0, 1\}, \ \forall j \in \mathbf{x}$$

where $\mathbf{c}$ is the real-valued vector containing the weighted power of choosing a given allocation, $\mathbf{x}$ is the vector of allocation selections, $\mathbf{A}_{\text{eq}}$ is a binary equality constraint matrix of K rows and $\mathbf{A}$ is a binary inequality constraint matrix of M rows. Each non-zero entry of the solution vector $\mathbf{x}$ corresponds to selecting the corresponding column allocation in $\mathbf{A}$.

The matrix $\mathbf{A}$ describes the set of potential RB allocations for all users. It is comprised of individual allocations given as

$$\mathbf{A} = [\mathbf{A}_1, \ldots, \mathbf{A}_K] \tag{33}$$

where $\mathbf{A}_i$ is a matrix containing the set of feasible allocations for UE i. Each column of $\mathbf{A}_i$ corresponds to a feasible allocation while each row corresponds to a specific resource. Each entry in $\mathbf{A}_i$ can take a value of $\{0, 1\}$. An entry of 1 if the particular resource is required by a UE for that allocation and 0 otherwise.

The set of possible allocations is determined as follows for each UE. During any subframe, a UE with data for transmission will utilize between one and M RBs in frequency. Any unique possible allocation is given as a column entry in $\mathbf{A}_i$. For example in the case where $M = 4$:

$$\mathbf{A}_i = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}. \tag{34}$$

The effective MCS scheme is a function of the number of RBs and T_i. For each possible contiguous block of RBs of size, the power level needed to maintain BLER_{tgt} for all possible allocations of contiguous resource blocks is found using (30). For each such possible contiguous block above given by columns in $\mathbf{A}_i$, the corresponding transmission power is given in the corresponding entry of $\mathbf{c}$.

The equality matrix $\mathbf{A}_{\text{eq}}$ is simply a matrix of K rows constraining the number of selected allocations such that each UE is only allotted one allocation selection from their matrix $\mathbf{A}_i$. This is given as

$$\mathbf{A}_{\text{eq}} = \begin{bmatrix} \mathbf{1}_{C_1}^T & \cdots & \mathbf{0}_{C_K}^T \\ \vdots & \ddots & \vdots \\ \mathbf{0}_{C_1}^T & \cdots & \mathbf{1}_{C_K}^T \end{bmatrix} \tag{35}$$

where C_i is the number of columns in $\mathbf{A}_i$ and $\mathbf{1}_x$ and $\mathbf{0}_x$ are column vectors of length x.

The objective function vector $\mathbf{c}^T = [\mathbf{c}_1^T, \ldots, \mathbf{c}_K^T]$ is simply the cost of choosing the corresponding allocation for each $\mathbf{c}_i$. By the design of the problem, one can see the cost is simply the weighted power of choosing an allocation. Individual entries of $\mathbf{c}_i$ can be then be given as

$$c_{i:j_i} = \alpha_i P(\mathcal{N}_i(j_i), T_i), \quad j_i = 1, 2, \ldots, C_i \tag{36}$$

where the function $P_i(\cdot)$ is given in (30), α_i is the priority weight of UE i and where $\mathcal{N}_i(j_i) = \{x | a_{i:x,j_i} = 1, x = 1, 2, \ldots, M\}$. The quantity $a_{i:x,j_i}$ denotes the $\{x, j_i\}$ entry in $\mathbf{A}_i$ and j_i is the j_ith column of $\mathbf{A}_i$.

In Figs. 6 and 7 we show the result of the power allocation gap as a function of number of users and resource blocks. Here we set all users to have a required data rate of 400 bits per subframe with the same relative priority ($\alpha_i = 1, \forall i$). Fig. 7 is zoomed into a region of interest of Fig. 6. What we observe is that as long as the number of resource blocks is at least twice the number of users, the gap between the optimal and suboptimal allocation schemes is less than 10%. While this results in a relatively small increase in power allocated, there is a relatively large

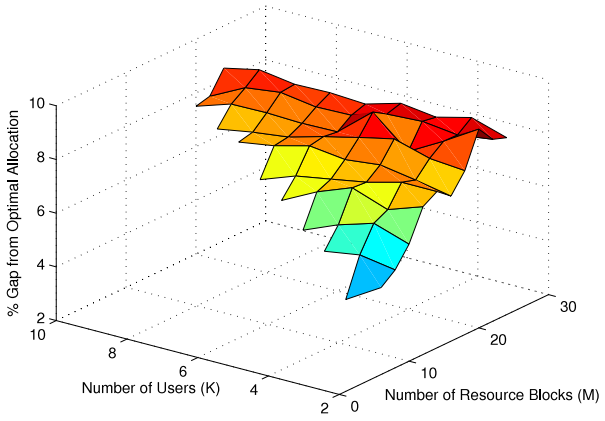
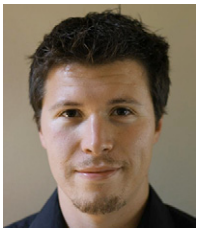


Fig. 7. Optimal allocation gap—zoom.

reduction in computational complexity, and the latter method can be easily implemented in real time.

References

[1] C.Y. Wong, R. Cheng, K. Lataief, R. Murch, Multiuser OFDM with adaptive subcarrier, bit, and power allocation, IEEE Journal on Selected Areas in Communications 17 (10) (1999) 1747–1758.
[2] M. Bohge, J. Gross, A. Wolisz, M. Meyer, Dynamic resource allocation in OFDM systems: an overview of cross-layer optimization principles and techniques, IEEE Network 21 (1) (2007) 53–59.
[3] D.J. Dechene, A. Shami, Energy efficient quality of service traffic scheduler for MIMO downlink SVD channels, IEEE Transactions on Wireless Communications 9 (12) (2010) 3750–3761.
[4] Q. Liu, S. Zhou, G. Giannakis, Queuing with adaptive modulation and coding over wireless links: cross-layer analysis and design, IEEE Transactions on Wireless Communications 4 (3) (2005) 1142–1153.
[5] X. Bai, A. Shami, Two dimensional cross-layer optimization for packet transmission over fading channel, IEEE Transactions on Wireless Communications 7 (10) (2008) 3813–3822.
[6] D.J. Dechene, A. Shami, Energy efficient resource allocation in SC-FDMA uplink with synchronous HARQ constraints, in: Proc. of IEEE International Conference on Communications, Kyoto, Japan, 2011.
[7] I. Wong, O. Oteri, W. Mccoy, Optimal resource allocation in uplink SC-FDMA systems, IEEE Transactions on Wireless Communications 8 (5) (2009) 2161–2165.
[8] D. Buckingham, Information-outage analysis of finite-length codes, Ph.D. Thesis, West Virginia University, 2008. http://wvuscholar.wvu.edu:8881/exlibris/dtl/d3_1/apache_media/13988.pdf.



Dan J. Dechene received his B.Eng. degree in Electrical and Computer Engineering from Lakehead University, Thunder Bay, ON, Canada in 2006, and his Masters and Ph.D. Degrees in Electrical and Computer Engineering from the University of Western Ontario, London, ON, Canada in 2008 and 2012 respectively with a research focus on energy efficient resource allocation techniques and multiple antenna systems. He is currently working as an Engineer at IBM's Semiconductor Research and Development Center in Hopewell Junction, NY. USA. Dr. Dechene is a member of IEEE.

Abdallah Shami received the B.E. degree in Electrical and Computer Engineering from the Lebanese University, Beirut, Lebanon in 1997, and the Ph.D. Degree in Electrical Engineering from the Graduate School and University Center, City University of New York, New York, NY in September 2002. In September 2002, he joined the Department of Electrical Engineering at Lakehead University, Thunder Bay, ON, Canada as an Assistant Professor. Since July 2004, he has been with *Western University*, Canada where he is currently an Associate Professor in the Department of Electrical and Computer Engineering. His current research interests are in the area of wireless/optical networking.

Dr. Shami is currently an Associate Editor for *IEEE Communications Letters* and the *International Journal of Communication Systems*. Dr. Shami has chaired key symposia for *IEEE GLOBECOM*, *IEEE ICC*, *IEEE ICNC*, and *ICCIT*. Dr. Shami is a Senior Member of IEEE.