A Heuristic for Fair Dynamic Resource Allocation in Overloaded OFDMA Systems

Adam N. Letchford* Qiang Ni† Zhaoyu Zhong*

To appear in *Journal of Heuristics*

Abstract

OFDMA is a popular coding scheme for mobile wireless communications. In OFDMA, one must allocate the available resources (bandwidth and power) dynamically, as user requests arrive and depart in a stochastic manner. Several exact and heuristic methods exist to do this, but they all perform poorly in the "over-loaded" case, in which the user demand is close to or exceeds the system capacity. To address this case, we present a dynamic local search heuristic. A particular feature of our heuristic is that it takes fairness into consideration. Simulations on realistic data show that our heuristic is fast enough to be used in real-time, and consistently delivers allocations of good quality.

Keywords: stochastic dynamic optimisation, local search, OFDMA systems, mobile wireless communications.

1 Introduction

In modern mobile wireless communication systems, the base stations often use a coding scheme called *Orthogonal Frequency-Division Multiple Access* or OFDMA (see, e.g., [3]). In OFDMA, there are a number of transmission channels, called *subcarriers*. At any given point in time, there is a set of *users* with known demands. Each subcarrier must be assigned to only one user, but a user may be assigned to more than one subcarrier. The data rate for each subcarrier is a nonlinear function of the power allocated to it, and there is a limited amount of power available.

There are actually many different optimisation problems associated with OFDMA systems, with various objective functions, side-constraints and planning horizons (see, e.g., [2, 9–14, 17, 19–21, 23–30]). Most of them have been shown to be $\mathcal{NP}$-hard [5, 15, 16].

*Department of Management Science, Lancaster University, Lancaster LA1 4YX. E-mail: {A.N.Letchford,z.zhong1}@lancaster.ac.uk

†School of Computing and Communications, Lancaster University, Lancaster LA1 4WA. E-mail: q.ni@lancaster.ac.uk

In our earlier paper [12], we considered a relatively simple problem, in which the set of users is treated as fixed. The problem is to allocate the power to the subcarriers, and the subcarriers to the users, in order to maximise the overall data rate, subject to satisfying the demand of each user. We called this the *joint subcarrier and power allocation problem with rate constraints* (SPARC), and presented an exact algorithm for it.

Now, let M be the maximum data rate achieveable by the system (in megabits per second, Mb/s), and let D be the total demand (again in Mb/s). When $D/M \leq 0.93$, the algorithm in [12] is very fast, taking only a fraction of a second. On the other hand, when $D/M > 0.93$, the algorithm becomes unacceptably slow, sometimes taking minutes to find a feasible solution (or prove infeasibility).

In this paper, we address the high-demand case. More precisely, let $D(t)$ denote the demand at time t. We are concerned with the situation in which $D(t)/M$ regularly exceeds 0.93. In this case, we say that the system is *overloaded*. It turns out that a very different approach is needed for the overloaded case. This is for several reasons:

1. At certain points in time, the system may not have the capacity to satisfy all of the users.

2. Thus, we may need to be content with only *partially* satisfying the users at certain times.

3. This in turn means that we must ensure that users are treated in a manner that is perceived to be *fair*.

4. Since an exact approach is likely to be too slow, we must use a *heuristic* approach.

5. To be of practical use, the heuristic must be able to *re-optimise quickly*, as users arrive and depart. (Equivalently, it must be suitable for a *stochastic dynamic* optimisation problem rather than a static one.)

To address these considerations, we devise a heuristic, based on the solution of a single small convex program, followed by the periodic application of local search. The neighbourhoods are specially designed so that we can search them within a fraction of a second. Extensive simulations on realistic data indicate that the heuristic is fast enough to be used in real-time, and consistently delivers allocations of very good quality (according to various quality measures).

We remark that our heuristic is most appropriate for so-called *non-delay-constrained* traffic (such as emails and file requests), for which occasional delays are acceptable. For *delay-constrained traffic* (such as phone calls and live video), our heuristic may be less useful. (See Tao *et al.* [25] for more on these two kinds of traffic.)

The paper is structured as follows. Section 2 contains a brief literature review. Section 3 describes the new problem in detail, and Section 4 describes the heuristic itself. The computational results are given in Section 5 and some final remarks are made in Section 6.

Throughout the paper, we let I denote the set of subcarriers. Each subcarrier $i \in I$ has a known *bandwidth* B_i (measured in MHz) and a known *noise power* N_i (in watts). The set of users at time t is denoted by $J(t)$. The demand of user $j \in J(t)$, in Mb/s, is denoted by d_j. The total demand at time t is denoted by $D(t)$. That is, $D(t) = \sum_{j \in J(t)} d_j$. When we are considering only a single time period, we drop the index t and just write J and D, respectively. The amount of power available, in watts, is denoted by P. We assume that the demand process is stationary, so that the expected value of $D(t)$ is constant over time. Borrowing from queueing theory parlance, we call the expected value of $D(t)/M$ the *traffic intensity* and denote it by ρ.

2 Literature Review

As mentioned in the introduction, there is by now an extensive literature on optimisation in OFDMA systems. For the sake of brevity, we review here only a few works of direct relevance.

Consider a single subcarrier $i \in I$. The classical *Shannon-Hartley theorem* [22] states that, if we allocate p watts of power to subcarrier i, then the maximum possible data rata achievable via subcarrier i, in bits per second, is

$$f_i(p) = B_i \log_2 \left(1 + \frac{p}{N_i}\right).$$

We remark that this function is concave (over the domain $\mathbb{R}_+$).

Now consider the case of multiple subcarriers, and recall that M denotes the maximum data rate achievable by the system. One can compute M quickly by solving the following NLP:

$$\max \left\{ \sum_{i \in I} f_i(p_{ij}) : \sum_{i \in I} p_i \leq P,\ p \in \mathbb{R}_+^{|I|} \right\}. \tag{1}$$

This NLP can be solved quickly using a method called *water filling* (see, e.g., [1, 3, 4]).

Now we recall the formulation of the SPARC presented in [12]. This formulation considers only a single time period. For each $i \in I$ and $j \in J$, let x_{ij} be a binary variable, taking the value 1 if and only if subcarrier i is assigned to user j, and let p_{ij} be a continuous variable, taking the value zero if $x_{ij} = 0$, but otherwise representing the amount of power supplied to subcarrier i. The SPARC is then formulated as the following mixed 0-1 convex program:

$$
\begin{array}{rlll}
\max & \sum_{i\in I}\sum_{j\in J} f_i(p_{ij}) & & \\
\text{s.t.} & \sum_{i\in I}\sum_{j\in J} p_{ij} \le P & & (2)\\
& \sum_{i\in I} f_i(p_{ij}) \ge d_j & (j\in J) & (3)\\
& \sum_{j\in J} x_{ij} \le 1 & (i\in I) & (4)\\
& p_{ij} \le P x_{ij} & (i\in I,\ j\in J) & (5)\\
& p_{ij} \in \mathbb{R}_+ & (i\in I,\ j\in J) & \\
& x_{ij} \in \{0,1\} & (i\in I,\ j\in J). &
\end{array}
$$

The constraint (2) imposes the limit on the total available power. The constraints (3) ensure quality of service (QoS). The constraints (4) ensure that each subcarrier is allocated to at most one user. The constraints (5), which are the *variable upper bounds*, ensure that x_{ij} takes the value 1 if $p_{ij} > 0$. The remaining constraints are self-explanatory.

As mentioned in the introduction, the algorithm in [12] works well when $D/M \le 0.93$, but is slow otherwise. Moreover, when $0.93 < D/M \le 1$, there is a chance that the SPARC is infeasible. We conclude that the approach in [12] is suitable only when (a) the user demands are more-or-less static, and/or (b) $D(t)/M$ rarely exceeds 0.93.

Finally, we mention that there is a stream of literature on *fairness* in multi-user communications systems (see, e.g., [2, 6–8, 17–19, 23, 24, 26]). As mentioned above, fairness will be relevant to us because, when the traffic intensity is high, we may not be able to satisfy the demands of all users.

3 A Stochastic Dynamic Version of the SPARC

It turns out that some thought is needed before one can formally define a stochastic dynamic version of the SPARC. In particular, one must consider (i) what constitutes an instance of the problem, and (ii) which function is to be optimised. These issues are covered in the following two subsections.

3.1 Instance data

As in the standard SPARC, we assume that the set of subcarriers I is fixed, and that we are given the bandwidths B_i, noise powers N_i, and power limit P. (In real-life systems, the N_i may fluctuate a little over time. Our approach can be extended to cover that case, but we do not give details, for brevity.) As for the users, we make the following assumptions:

- User arrivals are Markovian with known average rate λ (per second).
- The durations of the user requests are i.i.d., with known probability distribution and known mean $\bar{t}$ (in seconds).

- The user demands follow a known probability distribution with known mean $\bar{d}$ (in Mb/s).

One can check that, at steady-state, the expected number of users is $\lambda\,\bar{t}$ and the expected total user demand is $\bar{D} = \lambda\,\bar{t}\,\bar{d}$. For the traffic intensity, we have $\rho = \bar{D}/M$. In our preliminary experiments, we found that ρ is a reasonably reliable measure of the difficulty of an instance. (Other obvious potential drivers of difficulty are the variances of the user durations and user demands, but we did not find these to be so important in our simulations.)

3.2 Objective function

Some thought also needs to be paid to the objective function. In particular, one must address the issue of fairness mentioned in the introduction.

Now, let us temporarily consider the static case, in which all user demands are known. Let $p \in \mathbb{R}_+^{|I|\,|J|}$ be a fixed power allocation. For each user $j \in J$, we define the *user rate* $r_j = \sum_{i \in I} f_i(p_{ij})$ and the *satisfaction* $s_j = r_j/d_j$. Then, the demand of a user is met if and only if the satisfaction is at least one. A natural objective is then to maximise the mean of the s_j.

Unfortunately, the use of this "max-mean" objective can lead to very unfair solutions when the user demands have a wide range.

Example: Suppose that $|J| = 2$, and that the demands are 10 and 1. Suppose that $|I| = 3$, and the subcarriers have data rates of 8, 2 and 1, respectively. If we assign subcarriers 1 and 2 to user 1 and subcarrier 3 to user 3, the mean satisfaction will be 1. But if we assign only subcarrier 1 to user 1 and subcarriers 2 and 3 to user 3, the mean satisfaction will be 1.9. So the second vector is preferable according to the "max-mean" criterion, even though the first allocation completely satisfies both users. □

To avoid such unfair solutions, one could attempt to maximise the minimum satisfaction instead. Unfortunately, 'max-min' optimisation problems are notoriously difficult to solve, by either exact or heuristic methods, because the objective function is 'flat' (i.e., small changes in p may lead to no change in the value of the objective).

After some experimentation, we discovered the following alternative objective function:

Definition 1 *The "weighted harmonic mean satisfaction" (WHMS) is:*

$$\frac{D}{\sum_{j \in J} d_j s_j^{-1}} = \frac{D}{\sum_{j \in J} d_j^2/r_j}.$$

In our experience, maximising the WHMS tends to lead to solutions that perform very well according to the max-min criterion. Indeed, in the above example, the first solution has a WHMS of 11/(10+1) = 1, whereas the second (unfair) solution has a WHSM of $11/\left(\frac{100}{8}+\frac{1}{3}\right)=6/7$. The following proposition gives a partial explanation for this phenomenon.

Proposition 1 *Let $d \in \mathbb{R}_+^{|J|}$ be a demand vector and let R be a positive constant. Consider the following two continuous optimisation problems: the "max-min" problem*

$$\max\left\{\min_{j\in J}\{r_j/d_j\} : \sum_{j\in J} r_j = R,\, r_j > 0\,(j\in J)\right\}$$

and the "max WHMS" problem

$$\max\left\{\frac{D}{\sum_{j\in J} d_j^2/r_j} : \sum_{j\in J} r_j = R,\, r_j > 0\,(j\in J)\right\}.$$

These two problems have the same optimal solutions.

Proof. The solution to the max-min problem is to set r_j to d_jR/D for all j. This gives each user a satisfaction of R/D. Now, since D is fixed, the max WHMS problem is equivalent to

$$\min\left\{\sum_{j\in J} d_j^2/r_j : \sum_{j\in J} r_j = R,\, r_j > 0\,(j\in J)\right\}.$$

We solve this last problem using the method of Lagrange multipliers. We give the constraint $\sum_{j\in J} r_j = R$ a Lagrange multiplier λ and consider the Lagrangian

$$L(r,\lambda) \;=\; \sum_{j\in J} d_j^2/r_j \;+\; \lambda\left(\sum_{j\in J} r_j - R\right).$$

We now have

$$\partial L(r,\lambda)/\partial r_j \;=\; \lambda - d_j^2/r_j^2 \qquad\qquad (j\in J).$$

Setting these partial derivatives to zero, we obtain $d_j^2/r_j^2 = \lambda$ for all j, or, equivalently, $r_j = \sqrt{\lambda}/d_j$ for all j. Thus, the optimal r values are proportional to $1/d_j$. In other words, r_j is set to d_jR/D for all j, just as in the max-min solution. $\square$

Now let us return to the stochastic dynamic case. In light of the above, one might wish to compute a policy that maximises the *expected* WHMS, where the expectation is taken over an infinite number of time periods. Unfortunately, this looks like an extremely difficult task, especially in the overloaded case. So, as mentioned in the introduction, we content ourselves with a heuristic approach that updates the resource allocation in each time period.

4 The Heuristic

In this section, we present a heuristic for maximising the expected WHMS when the system is overloaded.

4.1 Initial solution

Before one can apply local search, one needs an initial solution to start from. To construct an initial solution, we use the greedy heuristic described in Algorithm 1. During the course of the algorithm, r_j is the current data rate given to user j. The choice of the factor of $\sqrt{d_j}$ is designed to make it more likely that channels with high data rate will be allocated to users with high demand, yet still ensure that at least some of the demand of each user is satisfied.

Algorithm 1: Greedy Constructive Heuristic

Input: bandwidths B_i, noise powers N_i, initial demands d_j.

Solve the NLP (1) and let p^* be the optimal solution;
Sort the channels in non-increasing order of $f_i(p_i^*)$ and let L be the sorted list;
for *each user* $j \in J$ **do**
  Set $r_j := 0$;
end
for *each channel* i *in the list* L **do**
  Assign channel i to the user with the smallest value of $r_j/\sqrt{d_j}$;
  (In case of ties, assign it to the user with highest d_j);
  Increase r_j by $f_i(p_i^*)$;
end
Output: Initial allocation of channels to users.

4.2 Local search

To improve the initial solution, we use a straighforward local search heuristic. This heuristic consists of two main phases, as described in Algorithms 2 and 3.

In the first phase, we take each subcarrier and check if it should be assign it to another user. This phase can be implemented to run in only $O(|I|\,|J|)$ time. Indeed, maximising the WHMS is equivalent to minimising

$$\sum_{j \in J} \frac{d_j^2}{\sum_{i \in I} f_i(p_{ij})}. \tag{6}$$

Algorithm 2: First Improvement Phase

Input: bandwidths B_i, noise powers N_i, demands d_j, fixed power allocation vector p^*, current subcarrier allocation, current data rates r_j.

for *each subcarrier* $i \in I$ **do**
  Let k be the user to which subcarrier i is currently allocated;
  for *each user* $j \in J \setminus \{k\}$ **do**
    if *the WHMS can be improved by re-allocating* i *to* j **then**
      Re-allocate subcarrier i to user j;
      Update r_k and r_j;
    end
  end
end

Output: Improved subcarrier allocation.

Algorithm 3: Second Improvement Phase

Input: bandwidths B_i, noise powers N_i, demands d_j, fixed power allocation vector p^*, current subcarrier allocation, current data rates r_j.

for *each pair of subcarriers* $\{i, i'\} \subset I$ **do**
  Let k, k' be the users to which the subcarriers are currently allocated;
  if $k \neq k'$ *and the WHMS can be improved by swapping the allocation of subcarriers* i *and* i' **then**
    Swap the allocation of subcarriers i and i';
    Update r_k and $r_{k'}$;
  end
end

Output: Improved subcarrier allocation.

If we take channel i and assign it to user j instead of user k, the function (6) will increase by

$$\frac{d_k^2}{r_k - p_i^*} - \frac{d_k^2}{r_k} + \frac{d_j^2}{r_j + p_i^*} - \frac{d_j^2}{r_j}.$$

If this is negative, then we can accept the proposed move. We can check this in constant time for a given i and j.

In the second phase, we take pairs of subcarriers and swap the users. This phase can be implemented to run in $O\big(|I|^2\big)$ time. Indeed, if subcarriers i and i' are assigned to users k and k', respectively, and we swap the allocation, the function (6) will increase by

$$\frac{d_k^2}{r_k + p_{i'}^* - p_i^*} - \frac{d_k^2}{r_k} - \frac{d_{k'}^2}{r_{k'} + p_i^* - p_{i'}^*} + \frac{d_{k'}^2}{r_{k'}}.$$

4.3 Extension to the dynamic case

To adapt our heuristic to the dynamic case, we basically run the local search heuristic periodically. Details are given in Algorithm 4. The key idea is that, if the set of users has changed, we restore feasibility and re-optimise as quickly as possible. In particular, we do not call Algorithms 2 and 3 more than once in any given time period. Then, with appropriate data structures, the time taken by the algorithm in each time period is only $O(|I|^2)$. This limit on the running time is necessary, since, in the real system, one needs to decide how to re-allocate the subcarriers in a fraction of a second.

5 Experiments

In this section, we report on some computational experiments that we conducted. The heuristic described in the previous section was coded in Julia v0.5 and run on an intel Core i7 3.1 GHz CPU, with 16GB of RAM, under Ubuntu 16.04.1 LTS. The program calls on MOSEK 7.1 (with default settings) to solve the initial NLP.

5.1 Test Instances

We took particular care to make our test instances as realistic as possible, based on the IEEE 802.16 standard. With regard to subcarriers, we set $|I| \in \{72, 180, 300\}$. The noise powers N_i are random numbers distributed uniformly in the open interval $(0, 10^{-10})$, and the bandwidths B_i are all set to 2.5MHz. As for users, we assume that (a) the inter-arrival times follow a negative exponential distribution, (b) the service times (in seconds) are uniformly distributed in $[1, 4]$, and (c) the demands d_j are obtained by sampling random numbers from a unit lognormal distribution, and multiplying by a positive constant c. Finally, the power limit P is set to $|I|/2$, in watts.

Algorithm 4: Dynamic Local Search Heuristic

Input: bandwidths B_i, noise powers N_i, initial set of users $J(0)$, initial demands d_j, number of time periods T.

```
Construct an initial solution using Algorithms 1, 2 and 3;
for t = 1, ..., T do
    Let J(t) be the current set of users;
    Let J⁻ = J(t − 1) \ J(t);
    if J⁻ ≠ ∅ then
        for j ∈ J⁻ do
            for each subcarrier that was allocated to user j do
                Re-allocate the subcarrier to an arbitrary user in J(t);
            end
        end
    end
    Let J⁰ be the set of users in J(t) that currently have no
     subcarriers allocated to them;
    if J⁰ ≠ ∅ then
        Let J⁺ contain all users in J(t) that currently have two or
         more subcarriers assigned to them;
        for j ∈ J⁰ do
            Let j⁺ be the user in J⁺ with the highest satisfaction;
            Let i be a subcarrier that was allocated to user j⁺;
            Re-allocate subcarrier i to user j;
            if user j⁺ now has only one subcarrier then
                Remove j⁺ from J⁺;
            end
        end
    end
    Re-optimise by calling Algorithms 2 and 3;
end
```

Note that, if the mean arrival rate (in users per second) is λ, then the expected number of users in the system at any given point in time is 2.5λ. Thus, we could control the expected number of users by varying λ. For $|I| = 72$, we considered three scenarios, in which the expected number of users is 4, 6 or 8. For $|I| = 180$, we set the expected number of users to 10, 15 and 20. For $|I| = 300$, we set it 20, 30 and 40. This leads to nine scenarios in total (see the two left-most columns in Table 1).

In a similar manner, by careful selection of the scaling constant c, we could implicitly control the traffic intensity ρ. We considered four different values for c, corresponding to setting $\rho \in \{0.90, 0.95, 1.00, 1.05\}$. This means a total of 36 simulations. Each simulation was run for 1100 time periods, where the first 100 were used to allow the system to settle into steady state. So, we view T as being equal to 1000 in what follows.

5.2 Results

Recall that $J(t)$ denotes the set of users at time t and s_j denotes the satisfaction of user j. We first considered the following three performance measures.

- The *mean-min* satisfaction $\frac{1}{T}\sum_{t=1}^{T} \min_{j\in J(t)}\{s_j\}$.
- The *mean-mean* satisfaction $\frac{1}{T}\sum_{t=1}^{T} \frac{1}{|J(t)|}\sum_{j\in J(t)} s_j$.
- The *mean-max* satisfaction $\frac{1}{T}\sum_{t=1}^{T} \max_{j\in J(t)}\{s_j\}$.

We will call these simply "min", "mean" and "max" in what follows.

Table 1 shows the values taken by these three performance measures for various values of $|I|$ and various (expected) values of $|J|$, when $\rho = 1$. The columns headed "phase 1" concern a version of the heuristic in which the second improvement phase was omitted. We see that the heuristic performs remarkably well, with values close to or exceeding 1 in all cases. Interestingly, the second improvement phase has little effect on the mean satisfaction, but it improves the fairness of the solutions noticably. Note also that all three performance measures improve as the number of subcarriers increases, but worsen slightly as the expected number of users increases.

Table 2 reports the average time taken by each of our two improvement phases (i.e., Algorithms 2 and 3) in *one* time period, for the same simulations that were used for Table 1. We see that, in most scenarios, the routine is extremely fast, taking less than 0.2 seconds. The exception is the case $|I| = 300$, for which phase 2 can take up to a second. This suggests that phase 2 may not be appropriate when one is dealing with a large base station.

Finally, we make some comments about the traffic intensity, ρ. As one might expect, changing the value of ρ affected all three performance measures. Interestingly, in all cases we tried, the net effect was simply to multiply each number in Table 1 by approximately $1/\rho$. As for running times,

$\lvert I\rvert$	$\lvert J\rvert$	Phase 1 min	Phase 1 mean	Phase 1 max	Phase 2 min	Phase 2 mean	Phase 2 max
72	4	1.111	1.132	1.154	1.132	1.132	1.135
	6	1.055	1.094	1.133	1.090	1.094	1.105
	8	1.012	1.065	1.118	1.055	1.065	1.083
180	10	1.021	1.047	1.074	1.047	1.047	1.049
	15	0.983	1.025	1.067	1.023	1.025	1.034
	20	0.961	1.017	1.075	1.012	1.017	1.038
300	20	0.986	1.019	1.053	1.019	1.019	1.021
	30	0.960	1.012	1.065	1.009	1.012	1.029
	40	0.939	1.008	1.079	1.002	1.009	1.041

Table 1: Average values of performance measures during simulation ($\rho = 1$)

$\lvert I\rvert$	$\lvert J\rvert$	Phase 1 min	Phase 1 mean	Phase 1 max	Phase 2 min	Phase 2 mean	Phase 2 max
72	4	0.00005	0.00049	0.00207	0.00572	0.00811	0.01699
	6	0.00005	0.00077	0.00258	0.00590	0.00897	0.01860
	8	0.00005	0.00102	0.00311	0.00561	0.00898	0.01304
180	10	0.00014	0.00371	0.01154	0.04673	0.08529	0.12025
	15	0.00134	0.00620	0.01530	0.07202	0.09597	0.15561
	20	0.00206	0.00933	0.02383	0.08515	0.10873	0.17201
300	20	0.00528	0.01797	0.07447	0.30598	0.38317	0.91077
	30	0.01075	0.03051	0.10703	0.34284	0.42251	0.96528
	40	0.01781	0.04402	0.14996	0.34618	0.44667	1.09267

Table 2: Average values of running time ($\rho = 1$)

varying ρ had no noticeable effect. (This is probably because the bottleneck of the algorithm is phase 2, whose running time, $O(|I|^2)$, does not depend on ρ.) For these reasons, and also for the sake of brevity, we do not report detailed results for different values of ρ. In any case, the main conclusion is that the performance of the heuristic is not very sensitive to the traffic intensity.

6 Conclusion

In this paper, we have considered how to allocate resources in an OFDMA system when (a) the system is overloaded (i.e., the expected demand is close to or higher than the system capacity), and (b) users arrive and depart every few seconds, in a stochastic manner. Since an exact approach for this case seems to be out of the question, we have proposed a dynamic local search heuristic. The computational results indicate that our heuristic consistently achieves allocations that are both efficient and fair.

References

[1] T. Cover & J. Thomas (1991) *Elements of Information Theory.* New York: Wiley.

[2] M. Ergen, S. Coleri & P. Varaiya (2003) QoS aware adaptive resource allocation techniques for fair scheduling in OFDMA based broadband wireless access systems. *IEEE Trans. Broad.*, 49, 362-370.

[3] K. Fazel & S. Kaiser (2008) *Multi-Carrier and Spread Spectrum Systems (2nd edn.).* Chichester: Wiley.

[4] S. Haykin (1994) *Communication Systems.* New York: Wiley.

[5] P.-H. Huang, Y. Gai, B. Krishnamachari & A. Sridharan (2010) Subcarrier allocation in multiuser OFDM systems: complexity and approximability. In *Proc. of the 2010 IEEE Wireless Communications and Networking Conference.*

[6] K. Illanko, M. Naeem, A. Anpalagan & D. Androutsos (2014) Frequency and power allocation for energy efficient OFDMA systems with proportional rate constraints. *IEEE Wirel. Commun. Lett.*, 3, 313–316.

[7] R. Jain, D.M. Chiu & W. Hawe (1984) A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. *Technical Report 301*, Digital Equipment Corporation, Hudson, MA.

[8] F. Kelly, A. Maulloo & D. Tan (1998) Rate control for communication networks: shadow prices, proportional fairness and stability. *J. Oper. Res. Soc.*, 49, 237–252.

[9] K. Kim, Y. Han & S.L. Kim (2005) Joint subcarrier and power allocation in uplink OFDMA systems. *IEEE Commun. Lett.*, 9, 526–528.

[10] D. Kivanc & H. Liu (2000) Subcarrier allocation and power control for OFDMA. In *Proc. of Asilomar Conf. on Signals, Systems and Computers*, vol. 1, pp. 147–151.

[11] X. Lei & Z. Liang (2015) Joint time-frequency-power resource allocation algorithm for OFDMA Systems. In *Proc. 5th Int. Conf. Electr. Info. Emerg. Commun.*, pp. 266–271.

[12] A.N. Letchford, Q. Ni & Z. Zhong (2017) An exact algorithm for a resource allocation problem in mobile wireless communications. *Comput. Optim. Appl.*, 68, 193–208.

[13] A.N. Letchford, Q. Ni & Z. Zhong (2017) Bi-perspective functions for mixed-integer fractional programs with indicator variables. *Working paper*, Lancaster University.

[14] A.N. Letchford, Q. Ni & Z. Zhong (2017) A heuristic for maximising energy efficiency in OFDMA systems with QoS constraints. *Working paper*, Lancaster University.

[15] Y.F. Liu & Y.H. Dai (2014) On the complexity of joint subcarrier and power allocation for multi-user OFDMA systems. *IEEE Trans. Signal Process.*, 62, 583–596.

[16] Z.Q. Luo & S. Zhang (2008) Dynamic spectrum management: Complexity and duality. *IEEE J. Sel. Topics Signal Process.*, 2, 57–73.

[17] W.V. F. Maurício, F.R.M. Lima, D.A. Sousa, T.F. Maciel & F.R.P. Cavalcanti (2016) Joint resource block assignment and power allocation problem for rate maximization with QoS guarantees in multiservice wireless systems. *J. Commun. Inf. Sys.*, 31, 211–223

[18] J. Mo & J. Walrand (2000) Fair end-to-end window-based congestion control. *IEEE/ACM Trans. Netw.*, 8, 556–567.

[19] M.K. Muller, S. Schwarz & M. Rupp (2013) QoS investigation of proportional fair scheduling in LTE networks. *Proc. 6th IFIP/IEEE Wireless Days Conference.*

[20] W. Rhee & J. Cioffi (2000) Increase in capacity of multiuser OFDM system using dynamic subchannel allocation. In *Proc. 2000 IEEE Veh. Technol. Conf.*, volume 2, pp. 1085–1089.

[21] K. Seong, M. Mohseni & J. Cioffi (2006) Optimal resource allocation for OFDMA downlink systems. *Proc. 2006 IEEE Inter. Symp. Inform. Th.*, 1394–1398.

[22] C. Shannon (1949) Communication in the presence of noise. *Proc. IRE*, 1, 10–21.

[23] Z. Shen, J.G. Andrews & B.L. Evans (2005) Adaptive resource allocation in multiuser OFDM systems with proportional rate constraints. *IEEE Trans. Wirel. Commun.*, 4, 2726–2737.

[24] Z. Song, Q. Ni, K. Navaie, S. Hou & S. Wu (2016) On the spectral-energy efficiency and rate fairness tradeoff in relay-aided cooperative OFDMA systems. *IEEE Trans. Wirel. Commun.*, 15, 6342–6355.

[25] M. Tao, Y.C. Liang & F. Zhang (2008) Resource allocation for delay differentiated traffic in multiuser OFDM systems. *IEEE Trans. Wirel. Commun.*, 7, 2190–2201.

[26] T. Ting, S.F. Chien, X.S. Yang & S. Lee (2014) Analysis of quality-of-service aware orthogonal frequency division multiple access system considering energy efficiency. *IET Commun.*, 8, 1947–1954.

[27] C.Y. Wong, R.S. Cheng, K.B. Lataief & R.D. Murch (1999) Multiuser OFDM with adaptive subcarrier, bit, and power allocation. *IEEE J. Sel. Areas Commun.*, 17, 1747–1758.

[28] X. Xiao, X. Tao & J. Lu (2013) QoS-aware energy-efficient radio resource scheduling in multi-user OFDMA systems. *IEEE Commun. Lett.*, 17, 75–78.

[29] C. Xiong, G. Li, S. Zhang, Y. Chen & S. Xu (2011) Energy-and-spectral-efficiency tradeoff in downlink OFDMA networks. *IEEE Trans. Wirel. Commun.*, 10, 3874–3886.

[30] C.C. Zarakovitis & Q. Ni (2016) Maximizing energy efficiency in multiuser multicarrier broadband wireless systems: convex relaxation and global optimization techniques. *IEEE Trans. Veh. Technol.*, 65, 5275–5286.