Uplink Resource Allocation Algorithms for Single-Carrier FDMA Systems

Fatma Irem Sokmen and Tolga Girici
Department of Electrical and Electronics Engineering
TOBB University of Economics and Technology
Ankara, Turkey
Email: {fsokmen, tgirici}@etu.edu.tr

***Abstract*—We have focused on SC-FDMA based resource allocation in uplink cellular systems. Subchannel and power allocation constraints specific to SC-FDMA are considered. We considered a binary integer programming-based solution recently proposed for weighted sum rate maximization and extended it to different problems. We considered problems such as rate constraint satisfaction with minimum number subchannels and sum-power minimization subject to rate constraints. Besides stating the binary integer programming formulations for these problems, we propose simpler greedy algorithms for the three problems. Numerical evaluations show that the greedy algorithms perform very close to the optimal solution, with much less computation time.**

I. Introduction

Over the past years Orthogonal Frequency Division Multiplexing (OFDM) has been an important technique for wideband wireless communications, and has been used in many wireless access systems such as WiMax and LTE. In OFDM, a wideband channel is divided into many orthogonal narrowband subcarriers, which provides resilience to multipath fading and intersymbol interference. However, OFDM also has high peak to average power ratio (PAPR) which requires expensive power amplifiers and reduces the energy efficiency of the system.

A technique called Single Carrier Frequency Division Multiplexing (SC-FDMA), which is an alternative to OFDM has the same overall performance as OFDM but with less PAPR [1]. In this technique each subcarrier is spread to the entire bandwidth and subcarriers are transmitted in series instead of parallel, which decreases the PAPR. This property is useful for mobile stations, since power is a precious resource for handheld mobile devices. Transmitting the subcarriers in series causes intersymbol interference, which requires frequency domain equalization [1], [2]. This can easily be done at the base station, so as a consequence SC-FDMA has been selected for Long-Term Evolution (LTE) *uplink* multiple access, and is an attraction to many system designers. In this paper we have considered joint subchannel and power allocation in uplink SC-FDMA for different problems like weighted rate maximization, minimum-subchannel resource allocation and minimum power resource allocation.

SC-FDMA is quite similar to OFDMA, such as the total bandwidth is divided into orthogonal subcarriers in order to be allocated to multiple users. As in OFDMA, in order to ease the resource allocation (it's harder to deal with subcarriers individually.) subcarriers are grouped into subchannels (chunks). There are two types of subchannelization methods in SC-FDMA: localized-FDMA and distributed-FDMA, where the subcarriers allocated to a user are either consecutive or distributed, respectively. L-FDMA takes advantage of frequency selective fading in maximizing the throughput. Distributed-FDMA avoids allocating adjacent subcarriers in deep fade [4] and provides resilience to frequency selective fading. Most of the previous works on SC-FDMA considers L-FDMA, as we do in this work.

In order to continue the single carrier property in multiple subchannels, the subchannels allocated to a user must be adjacent [3] [6]. This is an important resource allocation constraint specific to SC-FDMA. When power allocation is considered there are three restrictions; the total power transmitted by each user must be less than some maximum power level P^u, the power transmitted by each user on each sub channel should be less than some peak power level P^s and the power allocated to multiple sub channels for a user should be constant [3].

There are a number of resource allocation problems that have been investigated for SC-FDMA. In most of the papers authors have considered localized-FDMA subcarrier allocation however subchannel adjacency constraint was mostly not considered [4],[5], [7]. In [3] this constraint along with user and subchannel power constraints was considered in weighted rate maximization. This problem was formulated as a set partitioning problem, which was solved using binary integer programming. Then a much less complex greedy allocation algorithm has been proposed. The authors in [6] propose four greedy algorithms to obtain proportional fairness. In [6] power was never considered as a constraint or a resource allocation parameter. In [7] different types of greedy algorithms have been proposed based on a cost matrix. Most of the authors have preferred to use greedy algorithms which is more simple in solution, because its hard to deal with complex problems with many constraints. In the literature, generally weighted sum-rate maximization was considered and [3] is the only paper that considered all of the main constraints in SC-FDMA based resource allocation. In this paper we both improve the greedy algorithm in [3] and also formulate two more different practical problems and find both optimal and greedy algorithms to solve them.

We basically consider three different problems. First, we

978-1-4244-6001-4/10/$26.00 ©2010 IEEE

consider the proposed greedy algorithm in [3] and slightly improve its performance. Secondly, we consider the problem of resource allocation for minimum number of subchannels subject to user rate constraints. Using the framework in [3] we find the optimal solution. Again, we also propose suboptimal algorithms with comparable performance. Third, we consider resource allocation subject to rate constraints but this time with the objective of minimum total power expenditure. Again we propose optimal and efficient suboptimal solutions. We solve the maximum-throughput allocation problem and propose algorithms.

The paper is organized as follows; in Section II we explain the system model. In Section III, IV and V we formulate and solve the three different resource allocation problems. In each of these sections we state the problem, formulate the equivalent binary integer programming and propose some simpler greedy algorithms. In Section VII, using numerical evaluations we compare the performances of the optimal and suboptimal solutions for the problems. Section VII concludes the paper.

II. System Model

In this paper we consider a single cellular network with multiple users. We consider uplink Localized FDMA, where a number of adjacent subcarriers are grouped into subchannels. Hence, system bandwidth W is divided into a set $\mathcal{K} = \{1, \ldots, K\}$ of K orthogonal subchannels of bandwidth W_s each. There are a set $\mathcal{M} = \{1, \ldots, M\}$ of M users. Subchannel k of user m has channel gain $h_{m,k}$ and it is assumed to be known at the base station. Subchannel noise power at the receiver is $N_o W_s$ W/Hz. There are two kinds of power constraints , which are the user (P^u) and subchannel (P^s) power constraints. The base station acts as the scheduler; after certain calculations the base station sends the users the information about the resources that they are supposed to use. Since we consider three individual problems, our problem formulations change according to the target but the system models (above-mentioned parameters) and the optimization approach (proposed in [3] and extended to different problems here) remain the same. Let $\mathcal{K}_m$ be the set of subchannels allocated to user m and let $|\mathcal{K}_m|$ be the cardinality of it. The subchannels in this set must be adjacent in the frequency domain. Let p_m be the transmission power of user m, which is uniformly distributed to the $|\mathcal{K}_m|$ allocated subchannels. We assume the Shannon capacity formula as the relation between SNR and achievable rate. In this work , our main interest is the resource allocation algorithms and their relative performances. In practice, there may be imperfections due to power amplifiers in the user equipments and frequency domain equalization at the base station, however those are beyond the scope of this work.

III. Weighted Sum-Rate Maximization

The first problem we consider is exactly the same as the one studied in [3], which is a weighted-sum rate maximization problem. the user weights are denoted by w_m and reflect the priority of the user m. Here the peak subchannel power P^s and the user power constraint P^u should be utilized as much as possible to maximize rates. The problem formulation can be shown (1) which is a resource allocation problem with the SC-FDMA constraints. We follow the notation in [3] and repeat this problem formulation for the sake of completeness.

$$\max_{\mathcal{K}_1,\ldots,\mathcal{K}_M \in \mathbf{K}} \sum_{m\in\mathcal{M}} w_m \sum_{k\in\mathcal{K}_m} R_{m,k,\mathcal{K}_m} \; s.t K_m \cap K_{m'} = \emptyset, \quad \forall m \neq m', M \in \mathcal{M} \tag{1}$$

where $\mathbf{K}$ is the set of all allocations that satisfy the adjacency constraint, and $R_{m,k,\mathcal{K}_m}$ is the achievable rate by user m at one of its allocated subchannels k,

$$R_{m,k,\mathcal{K}_m} = \log_2\left(1 + \min\left(\frac{P^u}{|\mathcal{K}_m|}, P^s\right)\frac{h_{m,k}}{N_o W_s}\right), k \in \mathcal{K}_m \tag{2}$$

In [3] the problem is formulated as a binary integer programming. In order to do this , first the subchannel allocation pattern matrix is defined,

$$A_m = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}, \forall m \in \mathcal{M}$$

This matrix is the same for all users. A "one" means that the subchannel is allocated and a "zero" means the opposite. This is a $K \times C$ matrix, where C is the number of possible allocation patterns, which is equal to $\frac{1}{2}K^2 + \frac{1}{2}K + 1$ [3]. The adjacency constraint limits the possible allocation patterns, therefore makes the binary integer programming formulation possible. We define a resource allocation vector of size MC as $\mathbf{x} = [\mathbf{x}_1, \ldots, \mathbf{x}_M]^T$ and $\mathbf{x}_m = [x_{m,1}, \ldots, x_{m,C}]^T$. In this vector $x_{m,j} \in \{0, 1\}$, where a "one" corresponds to an allocation. For the weighted rate maximization problem, a reward vector $\mathbf{r}$ of the same size as $\mathbf{x}$ is defined, where $r_{m,j}$ is defined as

$$r_{m,j} = w_m \sum_{k\in\mathcal{K}_{m,j}} R_{m,k,\mathcal{K}_{m,j}} \tag{3}$$

where the set $\mathcal{K}_{m,j} = \{k \in \mathcal{K} : A_m(k,j) = 1\}$, set of subchannel indices corresponding to allocation pattern j. The reward vector becomes $\mathbf{r} = [\mathbf{r}_1, \ldots, \mathbf{r}_2]^T$ and $\mathbf{r}_m = [r_{1,1}, \ldots, r_{m,C}]^T$. The objective of the optimization is $\max_x \mathbf{r}^T\mathbf{x}$ and the constraints are $\mathbf{A}\mathbf{x} = \mathbf{1}_{M+K}$ and $x_j \in \{0, 1\}, \forall j$, where $\mathbf{1}_{M+K}$ is an $M + K$-length column vector. The matrix $\mathbf{A}$ is defined as

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \ldots & \mathbf{A}_M \\ \mathbf{1}_C^T & \ldots & \mathbf{0}_C^T \\ \vdots & \ddots & \vdots \\ \mathbf{0}_C^T & \ldots & \mathbf{1}_C^T \end{bmatrix} \tag{4}$$

where $\mathbf{1}_C^T$ is a row vector of all ones of length C. The top part of the equality, $[\mathbf{A}_1, \ldots, \mathbf{A}_M]\mathbf{x} = \mathbf{1}_K$,implies that each subchannel is allocated to a single user. The bottom part on the other hand, implies that each user selects only one of the patterns in $\mathbf{A}_m$. The MATLAB function `bintprog` is used in [3] to solve this problem. Of course, it takes unacceptably long time run this command, therefore a simpler algorithm is needed.

In [3] also a greedy algorithm was proposed for resource allocations. This algorithm allocates subchannels one by one to the user that best utilizes it. This algorithm is presented below in Algorithm 1, and its performance can be increased $7-8\%$ by a very simple modification. We named this algorithm as Maximum Utility Increase (MUI).

Algorithm 1 Greedy Algorithm: Maximum Utility Increase (MUI)

1. $r_m = 0$, $\mathcal{K}_m = \emptyset$ and $\mathcal{K}_m^f = \mathcal{K}, \forall m \in \mathcal{M}$, finish=0
2. MUI-1 [3]: Calculate $\Delta r_{m,k} = r_{m,\mathcal{K}_m \cup \{k\}} - r_m, \forall m \in \mathcal{M}, k \in \mathcal{K}$ using (3)
2. MUI-2: Calculate $\Delta r_{m,k} = \frac{1}{3}(r_{m,\mathcal{K}_m \cup \{k-1\}} + r_{m,\mathcal{K}_m \cup \{k\}} + r_{m,\mathcal{K}_m \cup \{k+1\}}) - r_m, \forall m \in \mathcal{M}, k \in \mathcal{K}$ using (3)

while $finish = 0$ **do**
  3. Find $[m^*, k^*] = \max_{m \in \mathcal{M}, k \in \mathcal{K}_m^f} \Delta r_{m,k}$
  if $\Delta r_{m^*,k^*} > 0$ **then**
    4. Set $\mathcal{K}_{m^*} = \mathcal{K}_{m^*} \cup k^*$, $\mathcal{K} = \mathcal{K} \setminus k^*$
    5. Calculate $r_{m^*} = r_{m^*,\mathcal{K}_{m^*}}$
    6. $\mathcal{K}_{m^*}^f = \{\min(\mathcal{K}_{m^*}) - 1, \max(\mathcal{K}_{m^*}) + 1\} \cap \mathcal{K}$
    7. $\Delta r_{m,k^*} = 0, \forall m$, $\Delta r_{m^*,k} = 0, \forall k$
    8. Calculate $\Delta r_{m^*,k} = r_{m^*,\mathcal{K}_m \cup \{k\}} - r_m^*, k \in \mathcal{K}_m^f$
  else
    9. finish=1;
  end if
  if $\mathcal{K} = \emptyset$ **then**
    10. finish=1;
  end if
end while

In Line 1 each weighted user rate r_m is set to zero, user subchannel allocation sets $\mathcal{K}_m$ are initialized as empty sets, and the feasible subchannel allocation set $\mathcal{K}_m^f$ is set to the subchannel set $\mathcal{K}$ which contains the whole subchannels. In Line 2, the increase of utility ($\Delta r_{m,k}$) by allocating a subchannel is calculated for all user-subchannel pairs. There are two variants of this algorithm; MUI-1 and MUI-2. In MUI-1, which is exactly the same method in [3] the $\Delta r_{m,k}$ values are computed as the difference between the utility when the subchannel k is added to $\mathcal{K}_m$ and current utility r_m. However, MUI-2 brings a small difference by taking into account the adjacent subchannels of subchannel k as well. This is done, because when we allocate a subchannel to a user, the user is limited to the neighbors of that channel because of the adjacency constraint. Of course, as exceptions, for $k = 1$ (respectively $k = K$) $k = 0$ (respectively $k = K+1$) does not exist. After calculating $\Delta r_{m,k}$ the following procedures goes the same for both methods. The user m^* and the subchannel k^* that maximizes $\Delta r_{m,k}$ is calculated in Line 3. If this value $\Delta r_{m,k}$ is greater than zero (i.e. if there is an improvement), the subchannel k^* is added to the set $\mathcal{K}_{m^*}$ and eliminated from the available subchannel set $\mathcal{K}$ (Line 4). In Line 5 the r_m^* is recalculated according to the new set $\mathcal{K}_{m^*}$. In Line 6 the feasible set of user m^* is calculated, as the user can only be given the left and right adjacent channels of $\mathcal{K}_m$, if they are available. The $\Delta r_{m,k^*}$ for all users are set to zero in order to avoid reallocation (Line 6). In Line 7, $\Delta r_{m^*,k}$ are recalculated considering user m^*'s feasible set $\mathcal{K}_{m^*}^f$. The algorithm executes until the available subchannel set $\mathcal{K}$ is empty.

IV. Minimizing number of used subchannels

The second problem formulated in (5), (6) and (7) is satisfying rate constraints R_m^0 for all users with minimum total number of subchannels. This problem has practical sense, because we would like to spend as fewer subchannels as possible for the rate constrained real time user such as voice and video and keep as many resources as possible for data applications demanding elastic traffic.

$$\min_{\mathcal{K}_1,\ldots,\mathcal{K}_M \in \mathbf{K}} \sum_{m \in \mathcal{M}} |\mathcal{K}_m| \tag{5}$$

s.t.

$$\sum_{k \in \mathcal{K}_m} \log_2 \left(1 + \min\left(\frac{P^u}{|\mathcal{K}_m|}, P^s \right) \frac{h_{m,k}}{N_o W_s} \right) > R_m^0, \forall m \in \mathcal{M} \tag{6}$$

$$K_m \cap K_{m'} = \emptyset, \forall m \neq m' \in \mathcal{M} \tag{7}$$

The objective is the total number of allocated subchannels and the constraints are the user rate and exclusivity constraints. We formulate a binary integer programming problem by defining the following $N \times C$ cost matrix.

$$s_{m,j} = \begin{cases} |\mathcal{K}_{m,j}| & \sum_{k \in \mathcal{K}_{m,j}} W_s \log_2 \left(1 + \frac{p_{m,k}^j h_{m,k}}{N_o W_s} \right) > R_m^0 \\ \infty & else \end{cases} \tag{8}$$

where $p_{m,k}^j = \min(\frac{P^u}{|\mathcal{K}_{m,j}|}, P^s)$. Here $s_{m,j}$ is the cost of the pattern j (simply the number of ones in the corresponding column of the matrix in (3)) for user m. If the pattern cannot satisfy the rate constraint for the user, the cost is infinity so that the allocation of the pattern for the particular user is discouraged[1]. An $MC \times 1$-sized column vector named $\mathbf{s}$ is formed as $\mathbf{s} = [\mathbf{s}_1, \ldots, \mathbf{s}_M]^T$ and $\mathbf{s}_m = [s_{1,1}, \ldots, x_{s,C}]^T$ and the following problem is formulated,

$$\min_{\mathbf{x}} \{\mathbf{s}^T \mathbf{x}\} \tag{9}$$

s.t.

$$[\mathbf{A}_1, \ldots, \mathbf{A}_M]\mathbf{x} \leq \mathbf{1}_K \tag{10}$$

$$\begin{bmatrix} \mathbf{1}_C^T & \ldots & \mathbf{0}_C^T \\ \vdots & \ddots & \vdots \\ \mathbf{0}_C^T & \ldots & \mathbf{1}_C^T \end{bmatrix} \mathbf{x} = \mathbf{1}_N \tag{11}$$

The constraint in (10) is an inequality because here a sub-channel doesn't have to be allocated. But a pattern has to be chosen for all users therefore (11) is an equality.

Alternatively, we propose a greedy algorithm called Block Allocation for Minimum Number of Subchannels (BMNS)

[1]The MATLAB function `bintprog` does not accept infinity, therefore a sufficiently high number such as K is given.

in Algorithm 2. The minimum-subchannel problem is quite different from rate maximization because each user has a rate requirement. Hence, finding a feasible allocation (where each user satisfies the rate requirements) is as important as spending minimum number of subchannels. The algorithms that are too *myopic* would end in unsuccessful allocations. For each user we allocate the subchannels *at once* as a block that is sufficient to satisfy the rate constraint. For this purpose we use the cost matrix defined in (8).

Algorithm 2 Greedy Algorithm: Block Allocation for Minimum Number of Subchannels (BMNS)

1. Calculate $\mathbf{s} = \{s_{m,j}, \forall m \in \mathcal{M}, j \in \mathcal{C}\}$ using (8).
2. $I_m = 0$, $\mathcal{K}_m = \emptyset, \forall m \in \mathcal{M}, finish = 0$

while $finish = 0$ **do**
   3. BMNS 1: $[s^*, m^*] = \max_{m s.t. I_m=0}\{s_m^2 - s_m^1\}$
   3. BMNS 2: $[s^*, m^*] = \max_{m s.t. I_m=0}\{s_m^1\}$
   if $s_{m^*}^1 < \infty$ **then**
      4. Find $j^* = \min_{j \in \mathcal{C}}\{s_{m^*,j}\}$,
      5. $\mathcal{K}_{m^*} = \mathcal{K}_{m^*} \cup k, \forall k \in \mathcal{K}_{m^*,j^*}$,
      6. $I_{m^*} = 1$
      7. Find all $j \in \mathcal{C}$ s.t. $\mathcal{K}_{m,j^*} \cap \mathcal{K}_{m,j} \neq \emptyset$, and set: $s_{m,j} = \infty, \forall m$
   else
      8. finish=1;
   end if
   if $I_m = 1, \forall m \in \mathcal{M}$ **or** $\bigcup_m \mathcal{K}_{m \in \mathcal{M}} = \mathcal{K}$ **then**
      9. finish=1;
   end if
end while
if $\sum_{k \in \mathcal{K}_m} W_s \log_2\left(1 + \frac{p h_{m,k}}{|\mathcal{K}_m| N_o W_s}\right) > R_m^0$ **then**
   10. successful=1;
else
   11. successful=0;
end if

Line 1 in the pseudocode calculates the costs $s_{m,j}$ for all user-pattern pairs using (8). In Line 2 the entries of the vector I_m are initialized to zero for all users $m \in \mathcal{M}$. I_m is an indicator value that takes value one if the user has already been allocated and avoids being allocated twice. The set of allocated subchannels $\mathcal{K}_m$ are initialized to empty sets for all users. The algorithm executes as long as all users or all subchannels are allocated or if it is understood that the allocation is infeasible. In Line 3 there are two approaches for selecting the user to be allocated among the users that have not been allocated before. These are the two variants of the algorithm, denoted as BMNS-1 and BMNS-2, respectively. Here s_m^1 is the smallest and s_m^2 is the second smallest cost for subchannels of node m. BMNS-1 chooses the user that maximizes the difference between the first and second smallest costs. This reflects the possible missed opportunity of not allocating the currently best feasible pattern for the user. BMNS-2 chooses the user that maximizes the minimum pattern cost. The maximizing user has possibly more urgency of allocation. If the minimum cost of the chosen user m^* is infinity, the allocation is infeasible and the algorithm no longer executes. If not, the available allocation pattern j^* minimizing the cost of m^* is given to this user and in Line 5 and line 6 the values I_{m^*} and $\mathcal{K}_{m^*}$ are updated accordingly. In Line 7, the allocation patterns that have common subchannels with the allocated pattern are given infinity cost, so that they are not reallocated (exclusivity constraint). The conditions for finishing the algorithm are checked in Line 9. After the algorithm ends, Lines 10 and 11 check if the algorithm successfully satisfies the rate constraints for all users.

V. Sum-power minimization

In our last problem we consider total power minimization subject to rate constraints. The motivation of this problem is minimizing the energy expenditure of the mobile devices in the uplink real time (e.g. telephone) transmission. The problem can be formulated as:

$$\min_{\mathcal{K}_1,\ldots,\mathcal{K}_M \in \mathbf{K}} \sum_{m \in \mathcal{M}} p_{m,\mathcal{K}_m} \tag{12}$$

s.t.

$$\sum_{k \in \mathcal{K}_m} \log_2\left(1 + \min\left(\frac{p_{m,\mathcal{K}_m}}{|\mathcal{K}_m|}, \frac{P^u}{|\mathcal{K}_m|}, P^s\right)\frac{h_{m,k}}{N_o W_s}\right) > R_m^0, \forall m \in \mathcal{M} \tag{13}$$

$$K_m \cap K_{m'} = \emptyset, \forall m \neq m' \in \mathcal{M} \tag{14}$$

$p_{m,\mathcal{K}_m}$ denotes the power allocated to a user m given an allocation $\mathcal{K}_m$, which is divided equally across its sub-carriers and sufficient to satisfy the rate constraint R_m^0,

$$p_{m,\mathcal{K}_m} = \min_p s.t. \sum_{k \in \mathcal{K}_m} W_s \log_2\left(1 + \frac{p h_{m,k}}{|\mathcal{K}_m| N_o W_s}\right) > R_m^0 \tag{15}$$

This value can be found using Newton's method in a few iterations.

This problem can also be formulated as a binary integer programming problem. For this purpose we define the cost matrix , where $e_{m,j}$ denotes the power expenditure user m if pattern j is used.

$$e_{m,j} = \begin{cases} p_{m,\mathcal{K}_{m,j}} & p_{m,\mathcal{K}_{m,j}} < P^u \; and \; \frac{p_{m,\mathcal{K}_{m,j}}}{|\mathcal{K}_{m,j}|} < P^s \\ \infty & else \end{cases} \quad m \in \mathcal{M}, j \in \mathcal{C} \tag{16}$$

Here if a particular pattern j does not satisfy the rate constraint with the available power , then the cost is set to infinity in order to discourge its allocation. Then an MC sized cost vector $\mathbf{e}$ is formed similar to the previous problems. Then the binary integer programming problem becomes $\min_{\mathbf{x}}\{\mathbf{e}^T\mathbf{x}\}$ subject to the same constraints as (10) and (11).

To find a sub-optimal solution two greedy algorithms are proposed. First we propose a greedy algorithm called maximum power decrease (MPD). This algorithm is similar in structure to the greedy MUI algorithm proposed in [3].

In Line 1, users powers are initialized to a very high value. The set of allocated subchannels $\mathcal{K}_m$ and the set of feasible subchannels $\mathcal{K}_m^f$ for each user are also initialized as before.

Algorithm 3 Greedy Algorithm 1:Maximum Power Decrease (MPD)

1. $p_m = \infty$, $\mathcal{K}_m = \emptyset$ and $\mathcal{K}_m^f = \mathcal{K}, \forall m \in \mathcal{M}$, finish=0
2. Calculate $\Delta p_{m,k} = p_m - p_{m,\mathcal{K}_m \cup \{k\}}, \forall m \in \mathcal{M}, k \in \mathcal{K}$ using (15)

while $finish = 0$ **do**
  3. Find $[m^*, k^*] = \max_{m \in \mathcal{M}, k \in \mathcal{K}_m^f} \Delta p_{m,k}$
  if $\Delta p_{m^*,k^*} > 0$ **then**
    4. Set $\mathcal{K}_{m^*} = \mathcal{K}_{m^*} \cup k^*$, $\mathcal{K} = \mathcal{K} \setminus k^*$
    5. Calculate $p_{m^*} = p_{m^*,\mathcal{K}_{m^*}}$
    6. $\mathcal{K}_{m^*}^f = \{\min(\mathcal{K}_{m^*}) - 1, \max(\mathcal{K}_{m^*}) + 1\} \cap \mathcal{K}$
    7. $\Delta p_{m,k^*} = 0, \forall m$, $\Delta p_{m^*,k} = 0, \forall k$
    8. Calculate $\Delta p_{m^*,k} = p_{m^*} - p_{m^*,\mathcal{K}_m \cup \{k\}}, \forall k \in \mathcal{K}_{m^*}^f$
  else
    9. finish=1;
  end if
  if $\mathcal{K} = \emptyset$ **then**
    10. finish=1;
  end if
end while
if $\sum_{k \in \mathcal{K}_m} W_s \log_2\left(1 + \frac{p h_{m,k}}{|\mathcal{K}_m| N_o W_s}\right) > R_m^0$ **and** $p_m < P^u$ **and** $\frac{p_m}{|\mathcal{K}_m|} < P^s, \forall m \in \mathcal{M}$ **then**
  11. successful=1;
else
  12. successful=0;
end if

Finally the change in powers $\Delta p_{m,k}$ are calculated in Line 2. $\Delta p_{m,k}$ is the difference between the currently allocated power p_m and the power allocation when the subchannel k is additionally allocated to the user. After these initializations the algorithm executes until subchannel set $\mathcal{K}$ is empty or no more improvement can be obtained. The subchannel-user pair $[k^*, m^*]$ that decreases the power the most is found in Line 3. After adding the subchannel k^* to the set $\mathcal{K}_{m^*}$ in Line 4, this subchannel is eliminated from the set $\mathcal{K}$. After these changes, the user power p_{m^*} is recalculated in Line 5 and the set $\mathcal{K}_m^f$ is reconstructed in Line 6 as it is done in the MUI algorithm. In Line 7 $\Delta p_{m,k^*}$ is equaled to zero for each user m, in order to avoid reallocation. $\Delta p_{m^*,k}$'s are recalculated in Line 8 for each subchannel m^*, according to the new feasible set. In Lines 11 and 12 the success of the algorithm (i.e. satisfaction of the rate requirements and power constraints) are checked. This algorithm acts as if there is no power constraint. This greatly simplifies the algorithm, on the other hand it results in more unsuccessful allocations as will be shown in the numerical evaluation results.

The second proposed algorithm is the Block Allocation for Minimum Total Power (BMTP), which is proposed in order to better take into account the power constraints. Minimum energy problem is a though problem because there is conflict among the users. In order to decrease the user power, we need to allocate more subchannels to a user, but then this leaves fewer subchannels to other users. In order to relieve the conflicts, the BMTP algorithms determines upper bounds on number subchannels for all users. BMTP uses the $M \times C$ power cost matrix $e_{m,j}, \forall j \in \mathcal{C}, \forall m \in \mathcal{M}$ defined in (16). Initializations are done as usual in Line 1 and cost matrix is formed in Line 2. If there is any user that does not have any finite cost pattern, then the solution is infeasible and the algorithm is finished in Line 7. If the problem is feasible, then lower bounds are calculated in Line 3, where the lower bound L_m is the minimum number of subchannels corresponding to a finite cost pattern according to (16) for user m. If the sum of lower bounds for all users is larger than the total number of subchannels K then the algorithm stops executing in Line 4. If not, then the upper bounds are calculated in Line 5. For each user, the upper bound is K minus the sum of lower bounds for all other users. Using these bounds restrict the search area, avoids searching unnecessary patterns and maintain feasibility.

Algorithm 4 Greedy Algorithm 2: Block Allocation for Minimum Total Power (BMTP)

1. $p_m = \infty$, $\mathcal{K}_m = \emptyset$ and $\mathcal{K}_m^f = \mathcal{K}, \forall m \in \mathcal{M}$, finish=0
2. Calculate $e_{m,j}, \forall m \in \mathcal{M}, j \in \mathcal{C}$ using (15) and (16)

if $\exists j, \forall m$ s.t. $e_{m,j} < \infty$ **then**
  3. Calculate lower bounds

$$L_m = \min_{j \in \mathcal{C}} \{|\mathcal{K}_{m,j}| : e_{m,j} < \infty\}, \forall m \in \mathcal{M} \tag{17}$$

  if $\sum_m L_m > K$ **then**
    4. finish=1
  else
    5.Upper bounds: $U_m = K - \sum_{m \neq n} L_n, \forall m \in \mathcal{M}$
    6. $e_{m,j} = \infty$, $\forall j$ s.t. $|\mathcal{K}_{m,j}| > U_m \; \forall m \in \mathcal{M}$
  end if
else
  7. finish=1
end if
while $finish = 0$ **do**
  8. $[e^*, m^*] = \max_{m s.t. I_m = 0}\{e_m^2 - e_m^1\}$
  if $e_{m^*}^1 < \infty$ **then**
    9. Find $j^* = \min_{j \in \mathcal{C}}\{e_{m^*,j}\}$,
    10. $\mathcal{K}_{m^*} = \mathcal{K}_{m^*} \cup k, \mathcal{K} = \mathcal{K} \setminus k \forall k \in \mathcal{K}_{m^*,j^*}$,
    11. $I_{m^*} = 1$
    12. Find all $j \in \mathcal{C}$ s.t. $\mathcal{K}_{m,j^*} \cap \mathcal{K}_{m,j} \neq \emptyset$, and set: $s_{m,j} = \infty, \forall m$
  else
    13. finish=1;
  end if
  if $I_m = 1, \forall m \in \mathcal{M}$ **or** $\bigcup_m \mathcal{K}_{m \in \mathcal{M}} = \mathcal{K}$ **then**
    14. finish=1;
  end if
end while
15. Determine the feasible sets $\mathcal{K}_m^f, \forall m \in \mathcal{M}$ and allocate the rest of the subchannels $\mathcal{K}$ according to the MPD algorithm above.
if $\sum_{k \in \mathcal{K}_m} W_s \log_2\left(1 + \frac{p h_{m,k}}{|\mathcal{K}_m| N_o W_s}\right) > R_m^0$ **and** $p_m < P^u$ **and** $\frac{p_m}{|\mathcal{K}_m|} < P^s, \forall m \in \mathcal{M}$ **then**
  16. successful=1;
else
  17. successful=0;
end if

The cost for patterns that are out of these bounds are equaled to infinity for all users (Line 6). After upper and lower bound calculations the algorithm executes until all users are allocated or there are no subchannels left to allocate. Then the maximum of the difference between the second minimum and the first minimum subchannel among all nodes is found in Line 8. Allocation to user m^* and necessary set and cost updates are performed in Lines 9, 10, 11 and 12. After these allocations, there may still be unallocated subchannels. More decrease in transmission powers may be obtained by allocating these. In Line 15, the feasible subchannels are determined as in MUI and MPD. These subchannels are allocated one by one until the subchannels finish or there can be no more improvement. The success of the algorithm is checked in Lines 16 and 17, in order to determine whether the resultant allocations violate the constraints or not.

VI. Numerical Evaluations

We consider a single cell with radius of 1000 meters and 10 users are transmitting to a base station. The users are uniformly distributed in the cellular area and the path loss is $31.5 + 37.6 \log_{10} d$ dB, where d is the distance to the base station in meters. We assume a fading Gaussian channel with -161dBm/Hz noise power spectral density. A Non-Line of Sight channel with Log-Normal shadowing standard deviation of 8dB is considered. Subchannel bandwidth is 180KHz, subchannel power constraint is 10mW and user power constraint is $P^u = 200$mW [8] [9]. We generate several sets of distance and fading instances and for each of these we perform allocations and record the resulting performance values. We plot the cumulative distributions of these values.

A. Maximum Weighted Sum-Rate

Figure 1 displays the empirical cumulative distribution functions of weighted sum rates corresponding to the optimal allocation (found by binary integer programming), greedy algorithm MUI-1 in [3] and our modified version MUI-2 that considers the adjacent subchannels in reward vector computation. The user weights w_m are uniformly distributed between zero and one at each optimization instant. The results show that although having the same complexity, our modification results in observable performance improvement and approaches to the performance of the optimal scheme. Results also prove that our algorithm achieves $7.5 - 8\%$ average performance improvement over MUI-2. There still exists room for improvement to reach the optimal weighted sum rate, however at the cost of more complex algorithms. At each step, our MATLAB code for the greedy algorithm only updates the rewards of the user-subchannel pairs that needs to be updated, therefore it results in almost 500 times less computation time then the binary integer programming.

B. Minimum Number of Subchannels

Figure 2 displays the empirical cdf's of the number of subchannels corresponding to the optimal allocation and the two variations of the greedy BMNS algorithm. We consider user

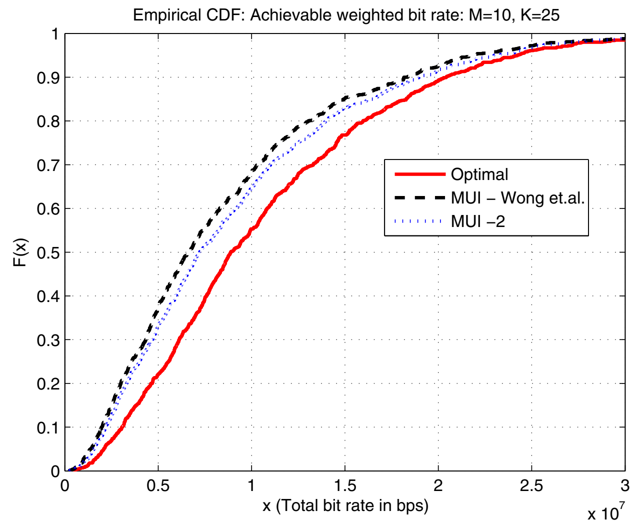

Fig. 1. Cumulative distribution of the weighted sum rate for different schemes. M=10 users and K=25 subchannels. The slight improvement that we made in the greedy algorithm of [3] achieves observable performance improvement (approximately 7 percent on average.)

rate constraints of 16kbps/user. BMNS-1 allocates the user that maximizes the difference of the second and first minimum pattern costs. BMNS-2 allocates the user that maximizes the minimum pattern cost. We look at 1000 cases and for the infeasible and unsuccessful allocations we give a high resultant cost in order to distinguish them. The numerical results show that the greedy algorithms perform almost identical to the binary integer programming solution. This is an important result because the greedy algorithms execute almost 30-35 times faster than MATLAB function `bintprog`. The optimal allocation is successful 51% of the time, while the greedy ones are 49% successful, with only 2 percent difference. Although the BMNS-1 algorithm that looks at the differential cost performs slightly better, BMNS-1 and 2 perform very similarly. We see that even for the optimal allocation, 49% of the times the allocation is infeasible because the log-normal shadowing and Rayleigh fading can take any number. In real time implementations, these algorithms should be used jointly with admission control and scheduling schemes, so that transmissions of users with bad channels are postponed. In this work, we are more interested in relative performances of the algorithms.

C. Minimum Sum-Power

In Figure 3 we see the cdf of the total power expenditure with the binary integer programming solution (top line) and the proposed greedy algorithms Maximum Power Decrease (MPD)(bottom) and Block Allocation for Minimum Total Power (BMTP) (middle). Since the bintprog function results in memory problems, we perform simulations for $K = 20$. The BMTP algorithm performs surprisingly close to the binary integer programming solution. The main reason of this is that we determine upper bounds for the number of subchannel allocations for all users. This avoids overallocation of some users

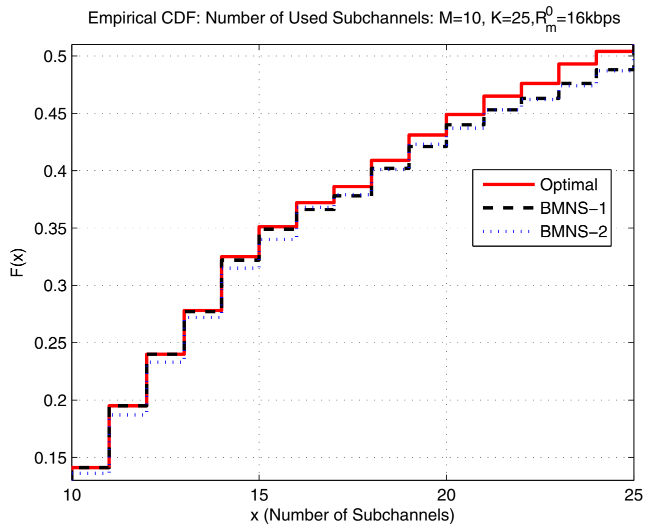

Fig. 2. Cumulative distribution of the weighted sum rate for different schemes. M=10 users and K=25 subchannels. Rate constraints are 16kbps per user. Proposed greedy algorithms perform almost as good as the binary integer programming solution.

and increases the chance of satisfying the rate constraints. We have also mentioned that MPD algorithm behaves as if there is no power constraint and as a result it has only 15% chance of successful allocation. The optimal and BMTP-based allocation have around 43% chance of successful allocation. The BMTP achieves this success at the cost of computational time. It can only achieve 10 times less computation time then the `bintprog` solution. MPD algorithm is 300 times faster than the `bintprog` function at the cost of poor performance.

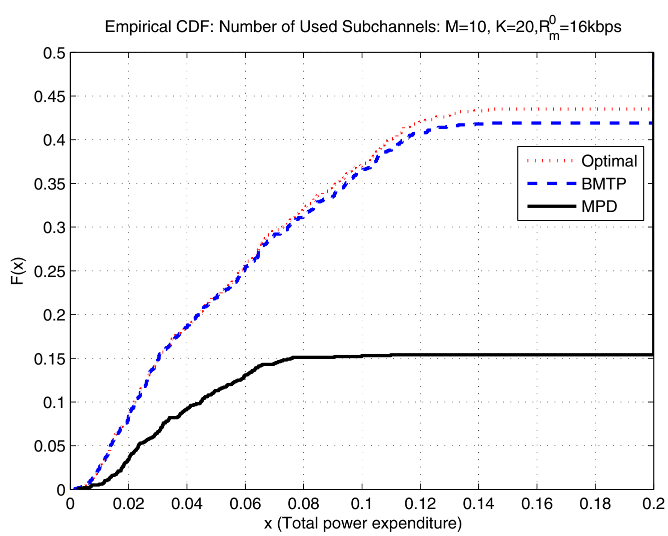

Fig. 3. Cumulative distribution of the weighted sum power for different schemes. M=10 users and K=20 subchannels. Rate constraints are 16kbps per user. Proposed greedy algorithm BMTP performs almost as good as the binary integer programming solution and it significantly outperforms the greedy MPD algorithm .

VII. Conclusions

We considered three resource allocation problems such as: weighted sum-rate maximization, transmission with minimal number of subchannels and sum-power minimization subject to rate constraints. For the first problem we made a simple modification to an algorithm proposed in the literature. The simulation results show that $7-8\%$ improvement can be typically obtained. For the latter two problems, we stated the problems as optimal binary integer programming problems, extending a framework proposed in the literature. We also proposed efficient greedy algorithms for these allocation problems. The numerical evaluation results show that for the minimum number of subchannel problem, the greedy algorithm works 30 times faster than the optimal solution and performs $97-98\%$ identical to it. As for the minimum sum-power problem we proposed an algorithm that runs at least 10 times faster than the optimal solution and performs almost identical to it.

Motivated by these promising results, first of all, the complexities of the greedy algorithms should be analyzed as future work. Possible less complex algorithms with close performance or better algorithms with acceptable complexity will be researched. Resource allocation for cooperative relayed transmission is also a direction of future research.

Acknowledgment

The work in this paper was supported by TUBİTAK under 1001 grant 108E208.

References

[1] Myung.H.G.,Lim J.,Goodman D.J., "Single Carrier FDMA for Uplink Wireless Transmission" IEEE Veh. Tech. Mag. september 2006
[2] Rumney M., "3GPP LTE: Introducing Single-Carrier FDMA", Agilent Measurement Journal, Inc., Issue 4, page 18, Jan. 2008
[3] Wong I.C., Oteri O., McCoy W. "Optimal Resource Allocation in SC-FDMA Systems " IEEE Trans. on Wireless Com. Vol. 8 No. 5. May 2009
[4] Lim J., Myung H.G., Oh K., Goodman D.J. "Channel Dependent Scheduling of Uplink Single Carrier FDMA Systems" IEEE 64th Veh. Tech. Conf., VTC-2006 Fall. 25-28 Sept. 2006 Page(s):1 - 5
[5] Lim J., Myung H.G., Oh K., Goodman D.J. "Proportional Fair Scheduling of Uplink Single-Carrier FDMA Systems" 17th annual IEEE symp. PIMRC 2006
[6] Lee S. Pefkianakis I., Meyerson A.,Xu S., Lu S. "Proportional Fair Frequency-Domain Packet Sceduling for 3GPP LTE Uplink" IEEE INFOCOM 2009
[7] Nwamadi O., Zhu X., Nandi A. "Dynamic Subcarrier Allocation For Single Carrier FDMA Systems", In Proc. of EUSIPCO 2008.
[8] Calabrese, F.D.; Michaelsen, P.H.; Rosa, C.; Anas, M.; Castellanos, C.U.; Villa, D.L.; Pedersen, K.I.; Mogensen, P.E., "Search-Tree Based Uplink Channel Aware Packet Scheduling for UTRAN LTE," IEEE Vehicular Technology Conference, 2008. , pp.1949-1953, 11-14 May 2008
[9] M. E. Ortega-Lerida, "Adaptive Radio Resource Management for VoIP and Data Traffic in 3GPP LTE Networks", Masters Thesis, Radio Communication Systems Group, KTH, Sweden, 2008