Throughput-constrained Scheduling in OFDMA Wireless Networks

Debalina Ghosh
Department of Computer Science
University of California at Davis, Davis, CA 95616
Email:debghosh@ucdavis.edu

Prasant Mohapatra
Department of Computer Science
University of California at Davis, Davis, CA 95616
Email:prasant@cs.ucdavis.edu

***Abstract*—Adaptive modulation and coding is an important characteristic of OFDMA based wireless networks. A group of subcarriers and symbols (which we refer to as "allocation unit") can be assigned to an user equipment (UE) based on its channel conditions. The "allocation units" can have different bandwidths for different UEs depending on the current channel conditions of the UEs. The UEs typically have a minimum throughput requirement. In this article our goal is to design a scheduler that maximizes the number of scheduled UEs while meeting their minimum throughput requirements. First we define an analytical model for scheduling and then propose three algorithms for the optimization problem. We show that, given the sets of "allocation units", maximizing the number of UEs is equivalent to finding the maximum independent set of a bounded degree graph. We also define the set allocation problem that minimizes the number of intersecting sets subject to certain constraints.**

I. Introduction

One of the most important features of Orthogonal Frequency Multiple Access (OFDMA) based wireless networks is adaptive modulation and coding. OFDMA divides the frequency into a number of subcarriers and divides time into symbols. The subcarriers may be combined into subchannels. Adaptive modulation and coding enables the different subcarriers to have different modulation and coding parameters for different user equipments (UEs) based on channel conditions. In IEEE 802.16 and IEEE 802.22 networks a slot is the smallest unit that can be allocated to any UE and it is identified by a symbol and subchannel combination. In 3GPP LTE the unit of allocation is a resource block that consists of 12 subcarriers. In this article we use the term "allocation unit"(AU) that can be a slot in IEEE 802.16 and IEEE 802.22 networks and a resource block in LTE networks.

We consider a centralized network model where there is a central entity and the UEs are connected to the central entity. We refer to this central entity as the base station (BS) from now on. The BS performs scheduling and assigns the "allocation units" to the UEs for a certain amount of time. Typically in wireless broadband networks, the UEs send a throughput request to the BS. The UEs also report the channel conditions periodically, which is used to determine the modulation and coding parameters. The UEs send channel quality indicator (CQI), which is an estimate of the downlink channel, to the BS. Based on these CQI reports, the base station assigns modulation and coding parameters to each of the "allocation units" for each UE. The BS then assigns the allocation units to the UEs based on their throughput requirement. An important goal of scheduling is to assign the allocation units in such a way that the maximum number of UEs can be scheduled. Note that this problem can be easily translated into an equivalent problem where applications have minimum rate requirements and hence the allocation units are assigned to applications instead of UEs. We use the term "UE" in this article that can be equally applied to applications instead.

Another emerging technology in wireless networks is the use of Multiple Input Multiple Output (MIMO) antennas. Single-user MIMO (SU-MIMO) allows only one UE to be scheduled over the same allocation unit in two possible modes. Multi-user MIMO (MU-MIMO) allows different UEs to be scheduled on different spatial streams over the same allocation unit. This allows the channel scheduler to have greater flexibility in scheduling.

The main contributions of our paper can be stated as follows:

- We formulate an analytical model of the problem of assigning "allocation units" to UEs such that the maximum number of UEs can be scheduled while satisfying the throughput requirements of the UEs. We extend this problem for SU-MIMO and MU-MIMO systems.
- We show the above problem is NP-hard by proving it is equivalent to the maximum independent set problem.
- We develop the "Greedy", "Min Links" and "Min Links with Set Allocation" algorithms for this problem.

The rest of the paper is organized as follows. Section II describes related work that has been done in this area. In Section III, we show that the problem is equivalent to the problem of finding the maximum independent set in a graph. We give a mathematical formulation of the problem in Section IV. We describe the "Greedy", the "MinLinks" and the "Min Links with Set Allocation" algorithms in Section V. Section VI presents some simulation results. Finally we conclude with a discussion of our work in Section VII.

978-1-4577-1379-8/12/$26.00 © 2012 IEEE

II. Related Work

Adoption of proportional fair (PF) algorithm to LTE networks is studied in [1] and [2]. The authors in [3] describe two approximation algorithms for frequency-domain packet scheduling in LTE uplink. The first algorithm is based on a greedy strategy whereas the second algorithm is based on the local ratio technique. The authors in [4] propose the clock-time proportional fairness (C-T PF) algorithm for the downlink of a multi-user MIMO system.

None of the above studies provide any guarantees on bandwidth. However [5] describes the Gradient algorithm with Minimum/Maximum Rate constraints (GMR) that optimizes a concave utility function $\sum_i H_i(R_i)$ of UE throughputs R_i, subject to certain lower and upper throughput bounds: $R_i^{min} \leq R_i \leq R_i^{max}$.

Tao, Liang and Zhang [8] investigate the resource allocation problem in a heterogenous multi-user OFDM system with both delay-constrained (DC) and non-delay-constrained (NDC) traffic.

Our work proposes a scheduler that provides minimum throughput guarantee in a time-varying and frequency-varying wireless network where multiple antennas might be present, whereas [5] considers only a time-varying channel. Note that our scheduler can be easily extended to provide a maximum throughput guarantee.

III. Problem Analysis

In this section, we show that the problem of maximizing the total number of scheduled UEs while satisfying their throughput requirements is equivalent to the maximum independent set problem.

A. Independent Set

An independent set is a set of non-adjacent vertices in a graph. Thus an independent set consists of l vertices such that there is no edge connecting any two vertices. A *maximum independent set* is a largest independent set for a given graph G. Finding the maximum independent set of a graph is an NP-hard optimization problem. Approximation algorithms for finding maximum independent sets are described in [10] and [11].

B. Equivalence of the Maximum Independent Set and the Maximum UE Assignment Problem

The maximum UE assignment problem assigns allocation units to UEs in such a way that

1) The throughput requirements of the UEs are satisfied.
2) No allocation unit is assigned simultaneously to more than one interfering UE using the same antenna.
3) The maximum number of UEs can be scheduled.

We first compute the Power Set of the set of allocation units A. Thus every element of the Power Set that satisfy the throughput requirement of a UE is a potential candidate for that UE. We construct a graph $G = (V, E)$ where V is identified by u, A_u where u identifies the UE and $A_u = a_1, a_2, \ldots a_i, \ldots a_n$ is a set of AUs that satisfies the throughput requirement of u. Thus if UE i has n_i sets that satisfy the throughput requirements, then there are n_i vertices in the graph G associated with UE i. Each vertex has edges connecting it to all vertices that are labelled with the same UE and all vertices that have a common allocation unit and are labelled with an interfering UE label. By our construction, a set of AUs corresponding to a vertex satisfies the throughput requirement of the UE corresponding to that vertex. Also, if two vertices do not have an edge connecting them, it means that they do not have any common AU or they belong to noninterfering UEs. Thus, finding out the maximum independent set of this graph gives us the maximum number of UEs that can be scheduled simultaneously.

The maximum independent set is a NP-hard optimization problem [12]. Hence no exact solution can be found in polynomial time. Our goal here is to develop approximation algorithms for this problem.

If a is the number of allocation units, the number of elements in a Power Set becomes 2^a. It is thus computationally prohibitive to find all the elements of a Power Set and hence our algorithms use a different approach to find the sets of AUs that satisfy the throughput requirements of the UEs.

IV. Mathematical Formulation

A. Model

Our system consists of a central entity (Base Station in IEEE 802.16 and IEEE 802.22 networks and eNB in LTE). UEs periodically report the current channel conditions to the central scheduling entity. As the channel conditions may be different for different UEs, each allocation unit has a different bandwidth for each UE depending on the modulation and coding paramaters. Let there be N UEs, where UE i is denoted by u_i. There are A allocation units, where a_j denotes the jth allocation unit. There are m antennas(modes). The bandwidth of each allocation unit varies from UE to UE and b_{im}^{a} denotes the bandwidth of allocation unit a for UE i and mode m. Each UE has a minimum throughput requirement denoted by R_i. X_{im}^{a} is an indicator variable that indicates whether allocation unit a is assigned to UE i using mode m. BW_i denotes the bandwidth actually allocated to UE i. I_i indicates whether UE i is scheduled or not. $NE(i)$ is the set of interfering neighbors of UE i.

B. Integer Linear Program

The optimization problem where each UE has m antennas is presented below. The optimization problem with only 1 antenna is similar and not shown here due to space constraints. We assume a UE will use the same MIMO mode m for all the allocation units for a scheduling period. This restriction can be easily removed.

$$
\begin{aligned}
& maximize \sum_i I_i \\
& \text{subject to the following constraints} \\
& R_i * I_i \leq BW_i \forall i \\
& BW_i * I_i = \sum_a (X^a_{im} * b^a_{im}) \forall i, m \\
& X^a_{im} + X^a_{jm} \leq 1 \forall a, \forall i, \forall j \in NE(i) \\
& X^a_{im} + X^a_{j\acute{m}} \leq M \forall a, \forall i, \forall j \in NE(i), \forall \acute{m} \neq m \\
& X^a_{im} + X^{\acute{a}}_{i\acute{m}} \leq 1 \forall \acute{a} \neq a, \forall \acute{m} \neq m, \forall i \\
& X^a_{im} \in 0, 1 \forall a, i \\
& I_i \in 0, 1 \forall i
\end{aligned}
$$

We omit the time dimension in all the constraints. The first constraint says that the bandwidth allocated to a UE should be at least equal to the throughput request. The second constraint says the bandwidth allocated to a UE is the sum of the bandwidths of all the allocation units assigned to that unit. The third constraint states that if an allocation unit with mode m is assigned to UE i, it cannot be simultaneously assigned to any interfering UEs. The fourth constraint states maximum number of interfering UEs that can be assigned simultaneously to the same AU is equal to the number of MIMO modes. The fifth constraint states only one MIMO mode can be used per UE across all the allocation units.

V. Approximation Algorithms

As mentioned earlier, the maximum UE assignment problem is a NP-hard problem and hence we cannot find exact solution. We describe three solutions here - a "Greedy" algorithm , "Min Links" algorithm and the "Min Links with Set Allocation" algorithm. All the algorithms construct a graph $G = (V, E)$. A vertex v is represented by u, A_u tuple where u identifies the UE and A_u is a set of AUs that satisfy the throughput requirement of u.

A. Graph Construction

The graph construction process starts by sorting the AUs for each UE by decreasing bandwidth. The AUs are then assigned to an UE until the throughput requirement of UE is met. That is , $S_i = S_i \cup AU$ until $bw(S_i) \geq R_i$ where S_i is a set of AUs associated with UE i, $bw(S_i)$ is the total bandwidth of all the AUs in S_i and R_i is the throughput requirement of UE i. This i, S_i combination becomes a vertex in the graph G. We form edges between vertices that have the same UE label or vertices that share an AU in common between interfering UEs. Thus vertices v_i and v_j has an edge e_{ij} between them if $UE(v_i) = UE(v_j)$ or $S(v_i) \cap S(v_j) \neq \Phi$ and $v_j \in NE(v_i)$ where $UE(v)$ gives the UE label of vertex v, $S(v)$ denotes the set of AUs of vertex v and $NE(v)$ is a set of all users that interfere with v.

In case of systems with only one antenna, if we choose to include only one set per UE, edges will be formed between vertices with interfering UE labels and intersecting AU sets.

In case of MU-MIMO systems, there are m sets corresponding to m antennas (modes). We sort the allocation units for each UE and mode. Then we assign the allocation units to UEs as before. Thus the graph will have $m * n$ vertices, where m is the number of MIMO modes and n is the number of UEs. Each vertex is labelled with the corresponding UE and the MIMO mode. All vertices having the same UE label are connected to each other. Edges will be formed between vertices belonging to interfering users that use the same antenna and have at least one common element in their AU sets.

B. "Greedy" Algorithm

The "Greedy" algorithm orders the UE in increasing order of throughput requirement. It then assigns the first UE to a AU set and makes this UE, set a vertex in a graph G. It keeps on doing this for all the UEs. When scheduling the UEs, the algorithm schedules the UE with the minimum throughput requirement. The algorithm deletes all the adjacent vertices in G. The scheduling process goes on until there are no remaining unscheduled vertices in the graph. The pseudocode is shown below in Algorithm 1. The time complexity of this algorithm is $O(n^2 m^2)$ where n is the number of UEs and m is the number of AUs. This is the time required to find the adjacent vertices of a vertex v and build the edges between the vertices.

for *each UE* u **do**
  Let A_u be a sorted list of allocation units in order of decreasing bandwidth;
end
for *each UE* u **do**
  $Sum = 0$;
  $S_u = \Phi$;
  while $Sum < bw_u$ **do**
    $S_u = S_u \cup a$;
    $A_u = A_u - a$;
    $Sum = Sum + bw_a$;
  end
  Append vertex (u, S_u) to Graph G
end
for *each vertex* v **do**
  ADJ_v = FormAdjacencies(v);
end
for *each UE* u **do**
  $assignedFlag_u$ = FALSE;
end
totalUsers = 0;
Let L_u be a sorted list of UEs in order of increasing bandwidth/throughput requirement;
for *each UE* u *in* L_u **do**
  if $assignedFlag_u$ *= FALSE* **then**
    if *UE* u *is the UE label of any vertex* v *in Graph G* **then**
      totalUsers = totalUsers + 1;
      Delete all the vertices that are included in ADJ_v;
    end
  end
end
Output totalUsers and assigned allocation units;

Algorithm 1: "Greedy" algorithm

C. "Min Links" Algorithm

The "Min Links" algorithm uses the concept of maximum independent set. Our algorithm is based on a simple approximate algorithm proposed by Halldorsson and Radhakrishnan([10]) for bounded degree graphs that has an approximation bound of (δ +2)/3 where δ is the maximum degree of any vertex.

The "Min Links" algorithm uses the graph construction process described earlier and then schedules the UE whose corresponding vertex has the least number of edges. The adjacent vertices are then deleted and the process continues until no more vertices are left in the graph. The output of the algorithm gives the total UEs assigned and the corresponding vertices gives us the allocation units assigned to each UE.

Claim - The "Greedy" algorithm has a ratio bound of n.

Proof - The maximum optimal solution is n, the maximum number of UEs. The "Greedy" algorithm selects at least one UE, assuming that the total bandwidth of the network is at least equal to the maximum bandwidth requested by a UE. Hence the ratio bound of the algorithm is n.

Claim - The "Min Links" algorithm has a ratio bound of n.

Proof - The maximum value of the Optimal solution is n. Let δ be the maximum degree of the graph. By our construction δ is no more than $n-1$. The minimum number of UEs found by the "Min Links" algorithm will be $\frac{n}{\delta+1}$ if n is divisible by $\delta+1$. If n is not divisible by $\delta+1$ then the minimum number of UEs will be $\frac{n}{\delta+1}+1$. Hence the ratio bound of "Min Links" algorithm is $\delta+1$ or n.

D. Solution to MIMO problem

In a system with MU-MIMO technology, a UE can have multiple antennas (modes). We assume that a UE uses only one antenna in a scheduling period. The algorithms construct graphs using the technique discussed in Section V-A.

The MimoGreedy algorithm orders the UEs in order of increasing throughput requirement. At each iteration, the algorithm schedules an unassigned UE with the minimum throughput requirement. Among all the vertices corresponding to that UE, it chooses the vertex with the least number of edges, which automatically selects the mode. The adjacent vertices are then deleted.

The "Min Links" algorithm for MU-MIMO is similar to the "Greedy" algorithm. The only difference with the "Greedy" algorithm is that the "Min Links" algorithm always chooses the UE corresponding to the vertex with the minimum number of edges when scheduling. The mode is chosen automatically as a result of choosing the vertex.

We can show that the Greedy algorithm has a ratio bound of n/m and the Min Links algorithm has a ratio bound of $\frac{\delta+1}{m}$ where δ is the maximum degree of the graph.

E. "Min Links with Set Allocation" algorithm

From the above discussion of the "Min Links" algorithm, we see that its performance depends on the maximum degree of the graph G. By our construction, this in turn depends on the number of intersecting sets, that is, what is the maximum number of vertices with interfering UEs and sets that has some AU in common. The number of interfering UEs depend on the network topology. Hence, if we can minimize the maximum number of sets that have some AU in common, our graph will have the lowest degree and perform best.

Instance - N UEs, the throughput requirement of each UE R_u, A allocation units and the bandwidth of each AU for each UE bw_u^i.

Problem - Let c_i be the number of times an AU i is included in a set. Our goal is to

$$min(max_i c_i)$$

such that

$$\sum_i bw_u^i \geq R_u \forall u$$

The "Min Links with Set Allocation" algorithm tries to minimize the number of times an AU is included in a UE-set combination by associating a count with each Allocation Unit. The count keeps track of how many times an AU occurs in an UE-set combination. Initially the count is 0 for all the AUs. When an AU is included in the set of a UE, the count is incremented by one. In this algorithm, we first sort the AUs by increasing count and then by decreasing bandwidth. Thus, within the sorted list, AUs with count 0 will appear first and then AUs with count 1 and so on. For all AUs with a specific count, the AUs will be sorted by decreasing bandwidth. Then the algorithm assigns the consecutive AUs to the UE concerned until the throughput requirement of the UE is met. We started assigning AUs to UEs starting from the first UE. Other ways to assign AUs might improve the performance such as assigning AUs to UE with least throughput requirement or UE with least number of high bandwidth AUs.

F. Profit Assignment

Our goal in this article is to maximize the total number of scheduled UEs. In many cases, UEs might have a profit associated with them. For example, higher paying UEs can have a higher profit associated with them than lower paying UEs or UEs with already existing real-time applications can be assigned a higher profit than UEs with best-effort applications.

Adaptive modulation and coding in different "allocation units" enables OFDMA to deal with the variable nature of wireless networks as each UE can be assigned a different rate depending on the channel conditions. A fundamental physical layer characteristic of OFDMA is that the channel conditions are correlated in both time and frequency. Hence if an UE i has good channel conditions in AU a in frame f, it is likely to have good channel conditions in the neighboring AUs. Our Maximum UE Allocation problem where all the profits are set to 1, tries to take advantage of this characteristic of wireless networks as when the number of UEs are maximized for a frame, the UEs with the best channel conditions are most likely to be selected.

However if the channel conditions are fairly stable for a number of frames, some UEs might get starved. To deal with that, we can use some fairness measure. One of the most common fairness measure used in wireless networks is proportional fairness. In proportional fairness, UEs with the largest $\frac{d_i(t)}{T_i(t)}$ is chosen at each time instant where $d_i(t)$ is

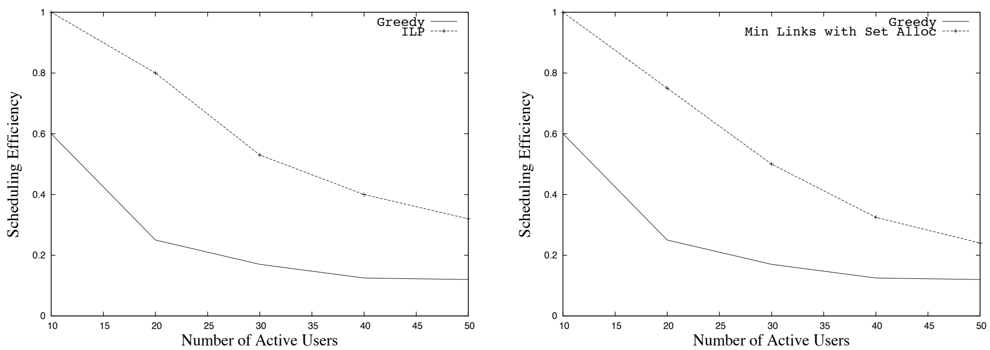

Fig. 1. Performance of ILP and Min Links with Set Allocation (1 antenna)

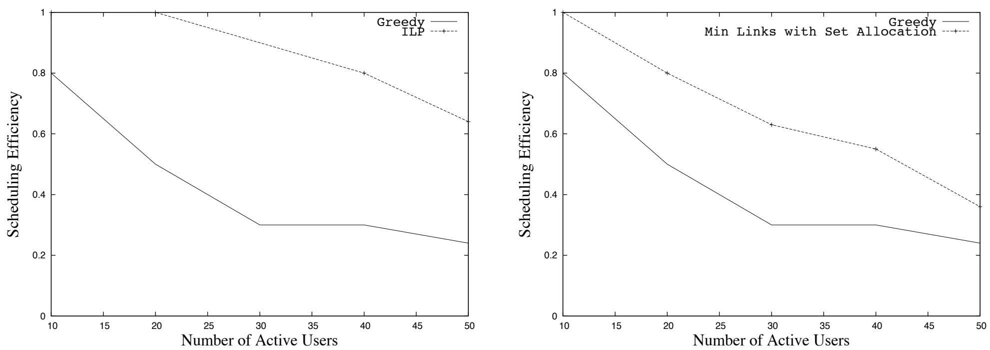

Fig. 2. Performance of ILP and Min Links with Set Allocation (2 antennas)

the data rate achievable by this UE at time t and $T_i(t)$ is the throughput of UE till t. Thus proportional fairness gives priority to UEs with high channel rate and low current average service rate. Another way to incorporate fairness is to make the profit $\frac{w_i}{F_i}$ where w_i is the weight associated with the UE and $F_i = 1+$ the number of frames where this UE has been already scheduled.

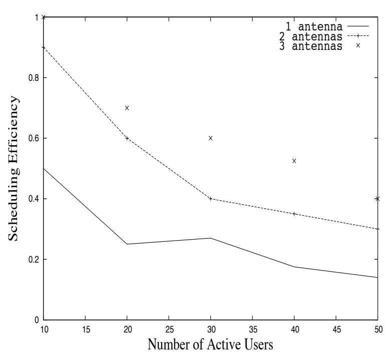

Fig. 3. “Min Links” algorithm: Number of scheduled UEs vs. number of active UEs with increasing antennas

VI. SIMULATIONS

In order to evaluate the performance of the proposed algorithms, we conducted simulations based on 3GPP LTE Model. A 3GPP LTE network consists of a central entity called eNB that assigns resource blocks (RBs) to UEs. A RB is comprised of a certain number of subcarriers. We assumed that the system bandwidth is 20 MHz, the subcarriers per RB is 12 and the RB bandwidth is 180 kHz. The number of RBs is 96. There are three different modulation/coding rate settings.

We use ILOG CPLEX 10.0 for modeling and solving the mathematical formulation of the problem. Although we performed simulations where UEs have varying throughput requirements, due to space constraints, we only show results where all the UEs have same throughput requirements here. We implement the algorithms using a custom simulator written in C. We varied the number of UEs from 10 to 50. We assume that the UEs are uniformly distributed in the cell. The modulation and coding parameters for each UE and AU pair is generated randomly. The scenarios shown here are for clique network topologies where all the nodes interfere with each other so that the simulation results depend only on how the AUs are selected.

We compare the algorithms based on ’Scheduling Efficiency’ metric where ’Scheduling Efficiency’ (SE) is the ratio of scheduled users to active users.

Figure 1 shows the results from the “Greedy”, the “Min Links” algorithm with Set Allocation and the ILP for UEs with only 1 antenna. When the number of UEs is less (10), both the ILP and Min Links with Set Allocation are able to schedule all the UEs, that is, scheduling efficiency (SE) is 1. The SE for all the algorithms decrease with the increase in the number of active users, as the throughput requirements of more number of UEs cannot be satisfied. Note that the bandwidth of the RBs and the throughput requirements of the

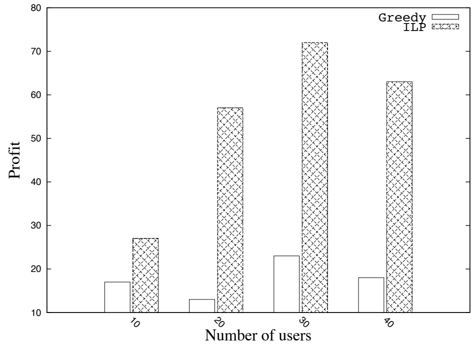

Fig. 4. Number of active UEs vs. the total profit of scheduled UEs

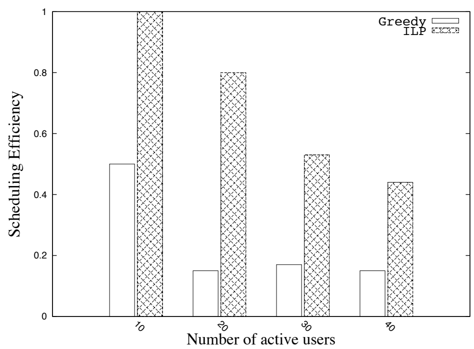

Fig. 5. Number of scheduled UEs vs. number of active UEs

UEs are generated randomly for each case. Hence, we can only compare the relative performance of the different algorithms at each point in the X axis. The ILP performs slightly better than the Min Links algorithm in the 1 antenna case; but, it performs considerably better than Min Links with Set Allocation when the number of antennas increases as can be seen fron Figure 2. The SE is 1 for the ILP when the number of UEs is less than 20 and the SE decreases to 0.65 for 50 UEs. In case of Min Links, the SE is 0.38 for 50 users.

Figure 3 shows the difference in the number of scheduled UEs when the number of antennas increases from 1 to 3. We only show results for "Min Links" algorithm as the results are similar in all the cases. Figure 4 compares the total profit of scheduled UEs for the ILP and the Greedy algorithm. Here, each UE has a weight that corresponds to the profit associated with the UE. The weight for each UE is chosen randomly from the set of integers 1 to 5.

As we see, the ILP performs significantly better than the Greedy algorithm. The answer can be found by examining Figure 5. The ILP schedules far more UEs than the Greedy algorithm. This is because when scheduling the Greedy algorithm just schedules the UE with the maximum profit without taking into consideration the number of neighbors of the UE. However, the ILP takes a very long time to produce the result and hence we omitted the result for 50 UEs. The ILP schedules roughly the same number of UEs for both the non-weight and weighted cases, because the total system bandwidth and the throughput requirements of the UEs are approximately the same in both the non-weight and weighted cases.

We do not show any results for Min Links algorithm for the weighted cases as a PTAS for the maximum weight independent set does not exist unless the weights are generated identically and independently distributed from a common distribution.

VII. Conclusion

In this article, we showed that the optimal assignment of throughput constrained UEs in OFDMA wireless networks is a NP-hard problem. We formulated an ILP for this problem and presented three algorithms. We did the same for OFDMA networks with MIMO with 2 or more antennas. However, our results show that the ILP performs better than the proposed algorithms whereas the running time of the ILP can be several hours. Our future work will focus on trying to find better approximation algorithms for this problem.

References

[1] S. Lee, S. Choudhury, A. Khoshnevis, S. Xu, and S. Lu, "Downlink mimo with frequency-domain packet scheduling for 3gpp lte," in *IEEE Infocom*, 2009.

[2] S. Lee, I. Pefkianakis, A. Meyerson, S. Xu, and S. Lu, "Proportional fair frequency-domain packet scheduling for 3gpp lte uplink," in *IEEE Infocom Mini-conference*, 2009.

[3] H. Yang, F. Ren, C. Lin, and J. Zhang, "Frequency-domain packet scheduling for 3gpp lte uplink," in *IEEE Infocom*, 2010.

[4] M. Torabzaden and W. Ajib, "Proportional fairness for mimo multi-user schedulers with traffic arrival process," in *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, 2008.

[5] M. Andrews, L. Quan, and A. Stolyar, "Optimal utility based multi-user throughput allocation subject to throughput constraints," in *IEEE Infocom*, 2005.

[6] K. L. C. Y. Wong, R. S. Cheng and R. D. Murch, "Multiuser ofdm with adaptive. subcarrier, bit, and power allocation," in *IEEE Journal on Selected Areas in Communications*, 1999.

[7] J. Jang and K. B. Lee, "Transmit power adaptation for multiuser ofdm systems," in *IEEE Journal on Selected Areas in Communications*, 2003.

[8] Y. L. M. Tao and F. Zhang, "Resource allocation for delay differentiated traffic in multiuser ofdm systems," in *IEEE Transactions on Wireless Communications*, 2008.

[9] R. Cohen and L. Katzir, "A generic quantitative approach to the scheduling of synchronous packets in shared uplink wireless channel," *IEEE Transactions on Networking.*, 2007.

[10] M. Halldorsson and J. Radhakrishnnan, "Greed is good: Approximating independent sets in sparse and bounded-degree graphs," *Algorithmica.*, 1997.

[11] P. Berman and M. Furer, "Approximating maximum independent set in bounded degree graphs," in *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1994.

[12] S. Dasgupta, C. Papadimitriou, and U. Vazirani, *Algorithms*. McGraw-Hill Science/Engineering/Math, 2006.

[13] T. Cormen, C. Leiserson, and R. Rivest, *Introduction to Algorithms*. Prentice Hall.

[14] D. Gamarnik, D. Goldberg, and T. Weber, "Ptas for maximum weight independent set problem with random weights in bounded degree graphs," in *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2010.