eOCSA: An Algorithm for Burst Mapping with Strict QoS Requirements in IEEE 802.16e Mobile WiMAX Networks

Chakchai So-In
Department of Computer Science & Engineering,
Washington University in St. Louis
St. Louis, MO 63130 USA
cs5@cse.wustl.edu

Raj Jain and Abdel-Karim Al Tamimi
Department of Computer Science & Engineering,
Washington University in St. Louis
St. Louis, MO 63130 USA
jain and aa7@cse.wustl.edu

***Abstract*— Mobile WiMAX systems based on the IEEE 802.16e standard require all downlink allocations to be mapped to a rectangular region in the two dimensional subcarrier-time map. Many published resource allocation schemes ignore this requirement. It is possible that the allocations when mapped to rectangular regions may exceed the capacity of the downlink frame, and the QoS of some flows may be violated. The rectangle mapping problem is a variation of the bin or strip packing problem, which is known to be NP-complete. In a previous paper, an algorithm called OCSA (One Column Striping with non-increasing Area first mapping) for rectangular mapping was introduced. In this paper, we propose an enhanced version of the algorithm. Similar to OCSA, the enhanced algorithm is also simple and fast to implement; however, eOCSA considers the allocation of an additional resource to ensure the QoS. eOCSA also avoids an enumeration process and so lowers the complexity to O(n^2).**

Keywords-component; Burst Mapping; Two-dimensional downlink mapping; Mobile WiMAX; IEEE 802.16e; Scheduling

I. INTRODUCTION

The IEEE 802.16e Mobile WiMAX standard [1] makes use of an Orthogonal Frequency Division Multiple Access (OFDMA) technique in order to achieve higher data rate, longer distance, and mobility. Basically, the entire channel is divided into multiple subcarriers. The number of subcarriers is proportional to the channel spectral width. These subcarriers are grouped into a number of subchannels. Then, each mobile station (MS) is assigned a group of subchannels for some OFDMA symbol times.

Mobile WiMAX uses a fixed frame-based allocation. Each frame is of 5 ms duration [2]. Bi-directional communication can be achieved by frequency division duplexing (FDD) in which uplink and downlink use different frequency bands or time division duplexing (TDD) in which the downlink (DL) traffic follows the uplink (UL) traffic in the time domain. In the FDD, the two subframes are parallel in time. For data traffic, the TDD provides a flexible partitioning of the frame into DL and UL subframes. Although we use a TDD system in this paper, the mapping algorithm we have introduced can be used for both systems.

The Mobile WiMAX frame starts with a downlink preamble and a frame control header (FCH) followed by the downlink map (DL-MAP) and the uplink map (UL-MAP). These maps contain the informational elements (IEs) that specify the burst profile for each burst. The profile consists of a burst-start time, a burst-end time, a modulation type, and a forward error control (FEC) used or to be used in the burst. Although we limit the discussion to one subscriber per burst, our algorithm can be easily applied to the case of multiple subscribers per burst, which is allowed by the standard.

In Mobile WiMAX systems, a base station (BS) has full control over resource allocations to various MSs in both the DL and the UL. In the DL, the BS decides the burst size based on the packets waiting in the queue to be sent to various subscribers. In the UL, the MSs send bandwidth requests for each connection that they have set up. Each connection has an agreed quality of service (QoS) requirement that is negotiated between the BS and the MS at the time of connection setup. The BS grants transmit opportunities to various MSs based on their bandwidth requests and QoSs.

Each UL data burst is allocated as a horizontal strip. The transmission starts at a particular slot and continues until the end of the UL subframe. After that, the allocation continues on the next subchannel. The horizontal allocation is used to minimize the number of subcarriers for each MS. This maximizes the power per subcarrier and hence the signal to noise ratio (SNR).

The IEEE 802.16e Mobile WiMAX standard requires that all DL data bursts be rectangular in shape. Although the standard allows more than one burst per subscriber, it increases the DL-MAP overhead. This particular case may be used when the subscriber really needs a different reliable channel as in different MCSs for different connections. The standard also allows more than one connection packing into one burst; however, the problem of rectangular mapping still remains.

Our algorithm primarily maps the resource for each subscriber into a downlink burst in a rectangular fashion. We do not consider more than one burst per subscriber. However, it is possible to pack multiple subscribers into one burst particularly if they are parts of the same physical mode. In this scenario, the unique connection identifier (CID) helps separate the subscribers. Packing multiple subscribers in one burst reduces the DL-MAP overhead [13], i.e., the DL-MAP IE size, and the proposed algorithm can be applied directly to this combined resource allocation. This rectangular criterion requires an efficient two-dimensional mapping algorithm. This is the main focus of this paper.

[1]This work was sponsored in part by grant from Application Working Group of WiMAX Forum

To assure QoS requirements, the downlink resource scheduling can be done in three steps. In the first step, before accepting new connections, the admission control module consults the scheduler to ensure that the required QoS of the new connection can be met. In this step, the scheduler computes the resource allocation (number of slots to be allocated) for that new connection and makes sure that allowing it will not violate the QoS for existing connections.

Then, for each frame the resource allocation can be done without any shape constraints and based solely on demand (the number of packets to be sent to/from a station), capacity (total available slots), and quality of service (QoS). Finally, in the downlink subframe this resource allocation is mapped in to the Mobile WiMAX frame in rectangular regions. At this step, the mapping module informs the scheduler and admission control modules if the resource allocation can be met without any QoS violations.

The two-dimensional rectangle mapping problem is a variation of the bin or strip packing problem, in which one is given bins to be filled with objects. The bin packing problems are known to be NP-complete [11]. The complexity of the solution grows exponentially with the number of objects or bins. There have been many attempts to overcome these problems as stated in [8, 9, 10]. However, there is no easy way to achieve the optimality with simple computation. Examples of simple approaches are to apply first-fit, next-fit, best-fit, or bottom-left allocation [8].

Moreover, many heuristic approaches have been introduced to alleviate the computational complexity, such as a level approach [8] is used to pack the fixed dimensional bins with non-increasing height from bottom to top and left to right, and then move to the next level when it reaches maximum allowable width.

The mapping problem in Mobile WiMAX systems is different from the original bin packing in that: first there are no fixed length and width limitations. Instead only bin sizes are given. Second, with increasing number of bursts (number of bins), the other end of the big bin (left side of the Mobile WiMAX frame) in which small bins are fitted also changes to allow increasing size of the variable part of DL_MAP.

In this paper, we propose a simple heuristic algorithm for the two-dimensional rectangular mapping for downlink bursts in IEEE 802.16e Mobile WiMAX with strict QoS requirements, that is, all resource allocations need to be mapped into the Mobile WiMAX frame with an additional resource consideration to ensure the QoS. We apply the concept of the largest area first and level mapping approach together. The paper is organized as follows: the two-dimensional rectangle mapping problem is revisited in Section II. Section III briefly describes some of the related works in a context of Mobile WiMAX systems. Our heuristic algorithm for the two-dimensional downlink burst mapping is described in Section IV. Then, the performance evaluation is presented in Section V. Finally, the conclusions are drawn in Section VI.

II. DOWNLINK BURST MAPPING PROBLEM STATEMENT

In IEEE 802.16e Mobile WiMAX, the two-dimensional downlink burst mapping can be stated as follows:

1) We are given a fixed rectangular bin B of width W and height H. The bin B has an area A equal to $W \times H$.
2) We are also given a set of n items $\{b_1, b_2, \ldots, b_n\}$. The ith item b_i has an area A_i
3) We need to *determin*e a rectangular shape for the ith item with width W_i and height H_i such that $\boldsymbol{A_i \leq W_i \times H_i}$.
4) Width $W_i \leq W$ for all i. Similarly, height $H_i \leq H$ for all i.
5) W_i, H_i, W, and H are all integers.
6) Since the mapped region is more than the desired allocation A_i, the extra resource is wasted and so, $W_i \times H_i - A_i$, should be minimized.
7) Due to the rectangular considerations, all n item bins may not fit into the big bin B; the goal is to minimize the additional resource width W that is required to fit all n item bins.

The unit of allocation in IEEE 802.16e Mobile WiMAX is "*slot*". The definition of slot depends upon the subchannelization mode and link direction (DL or UL) [1, 2].

In this paper, we assume the Partially Used Sub-Channelization (PUSC) mode, which is the most commonly used mode [2] in a mobile wireless environment. The analysis is applicable to other modes as well. The PUSC mode, the distributed subcarrier permutation, is suitable for mobile users when the channel condition is not constant over time and distance. The condition is averaged over the channel.

Consider downlink PUSC with 10 MHz channel. With WiMAX forum specified parameters [2], 10 MHz channel requires 1,024 subcarriers. In the downlink, these 1,024 subcarriers are grouped in to 30 subchannels with each subchannel consisting of fixed 28 subcarriers. A 5 ms frame and a 2:1 DL:UL ratio result in 13 slot columns in DL [12]. Thus, for this parameter set, the DL subframe consists of 14 slot columns and 30 rows resulting in a total of 420 slots. Of these, we allow 12 slot columns for QoS sensitive traffic. Rectangle mapping may require *one more slot column*. The remaining space is for MAPs and FCH.

Design Factors

As indicated earlier, the burst mapping problem is NP-complete [11]. A heuristic algorithm is proposed in this paper. A previous version of this algorithm was named OCSA or One Column Striping with non-increasing Area first mapping [3].

There are four main considerations in designing this algorithm. First, the algorithm should maximize throughput by minimizing unused space or unused slots, and by minimizing the extra space or over allocation slots, required to form rectangles. Second, the algorithm should be able to handle a large number of users and burst efficiently. Third, the algorithm should be aware of variable components of the DL-MAP and UL-MAP [12]. Finally, the algorithm should minimize the energy consumption of the mobile station.

III. RELATED WORK

In this section, we briefly restate some of the other burst mapping proposals for Mobile WiMAX. In [4 to 7], the algorithms basically map the resource allocations either row by row or column by column. In [4], full-search algorithms are limited to a maximum of 8 users. Each allocation is mapped in to multiple rectangles resulting in an increased DL-MAP overhead in [6]. In [7], a largest resource allocation size first criterion is also considered. However, the unused space is not considered or there is no detailed explanation of utilizing the unused space. The algorithm in [5] is similar to [7] but allows burst compaction.

The initial goal of OCSA [3] was to minimize the unused space. The algorithm mapped the resource allocation from bottom to top and right to left. All possible mapping-pairs were considered. In contrast, eOCSA considers only one best mapping-pair either the least width or height; therefore, eOCSA reduces the complexity to O(n^2).

In addition, OCSA did not consider the possibility of having extra resources to assure the QoS. In other words, OCSA is suitable for best effort traffic. Thus, eOCSA adds one more constraint in that the strict QoS requirements need to be assured. This may require additional resources. From our simulations, we found that on average only one more slot column is needed in order for the system to support all resource allocations.

IV. ENHANCED OCSA ALGORITHM

In this section, we describe our two-dimensional rectangular burst mapping algorithm. In general, eOCSA is similar to OCSA in that first to maximize the throughput, the resource allocations are sorted in a descending order (largest first). Then, the allocations are mapped from bottom to top and right to left to allow the space for the variable portions of the DL-MAP. Note that usually the allocation of the frame overheads i.e., DL-MAP is from top to bottom and from left to right. Third, the allocation favors lesser width to minimize energy consumption for mapping into the DL subframe. However, eOCSA does not consider all possible mapping pairs, but instead chooses only one pair which has the least width (vertical mapping) or height (horizontal mapping).

A. Algorithm Description

eOCSA consists of four steps as follows: First, given a set of resource allocations $\{A_i\}$, we sort the set in a descending order and select the largest element to map. Second step, *vertical mapping*, consists of mapping this resource allocation to the DL subframe. Given an area A_i, the algorithm maps the width-height pair (W_i, H_i) for the burst as follows:

$$W_i = \lceil A_i/H \rceil$$
$$H_i = \lceil A_i/W_i \rceil$$

Here, $\lceil\,\rceil$ denotes the ceiling function, and H is the maximum available height (DL subframe). With 10 MHz Mobile WiMAX, H is 30 subchannels. Note that this ensures that the mapped region is bigger than the required allocation ($W_i \times H_i \geq A_i$) and that the rectangle has the minimum possible width (minimizing MS active time and energy).

After a resource allocation is mapped to the DL subframe, some space may remain unallocated above the just mapped burst. In the third step, *horizontal mapping*, the eOCSA algorithm tries to assign this space (which we call a **strip**) to the next largest element, for example, *j*th allocation, that can be fitted in. In this step the region width is fixed, and it is used to determine the required height for the next largest element that can be fitted within this available region:

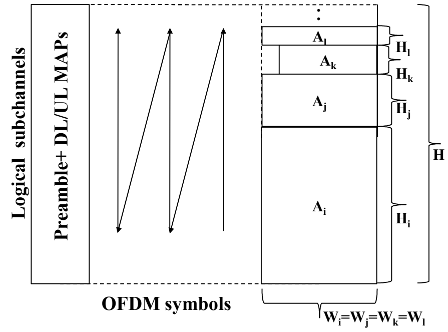

Fig. 1. An example of mapping downlink burst using eOCSA

Find largest A_j, such that $A_j < W_i \times H_0$

$$H_j = \lceil A_j/W_i \rceil$$
$$W_j = \lceil A_j/H_j \rceil$$

Here, $H_0 = H - H_i$ is the maximum available height in the strip. This step is repeated until either no space is left vertically, or there is no allocation that can be fitted in the available space. If no allocations can be found to fit, we move back to step 2 and select the next largest allocation to map in to the DL subframe. Fig. 1 shows the process of moving vertically and horizontally from right to left and bottom to top.

Fig. 2 shows a pseudo code for nesting of various steps of the eOCSA algorithm. Notice that the computational complexity in worst case of eOCSA is in the order of O(n^2), where n is the number of resource allocations within a frame.

$$Complexity = O(sorting) + O(allocation) = O(n \log n) + O(n^2)$$

Moreover, to achieve higher frame utilization, either a vertical or horizontal mapping step can be added to eOCSA with one more level of complexity.

```
Sorted_allocations = Sort (resource_allocations)                //1st step
FOR each unmapped element in sorted_allocations                 //2nd step
    Vertical_Mapping (&start_strip_i, &end_strip_i, &height_i)
    FOR each unmapped element in sorted_allocations             //3rd step
        Horizontal_Mapping (start_strip_i, end_strip_i, height_i,
        &sub_height_j)
    END FOR
END FOR
```

Fig. 2. Steps in eOCSA

Note that without additional columns' consideration; actually eOCSA can also roll the additional columns needed for the current frame to the next frame before beginning the next frame mapping. However, this may cause an extra delay. Moreover, without the extra columns a priority mechanism needs to be applied. For example, the resource allocation with the highest priority is moved to the beginning of the mapping queue thus being mapped regardless of the largest size consideration. However, this may lead to more unused space.

B. eOCSA Example

In this section, we provide an example that helps explain our algorithm. The main idea is to strictly map all resource allocations into a Mobile WiMAX frame to meet the QoS requirements.

In this example, the scheduler makes an allocation decision for ten mobile stations (MSs) in a Mobile WiMAX DL subframe. Table I shows a simple example for ten MSs randomly chosen. These MSs have been allocated A_1 through A_{10} by the scheduler as shown in the first row of the table. Basically, the sum of all resource allocations is 360 or 12×30.

TABLE I
TEN RANDOM RESOURCE ALLOCATIONS: EXAMPLE

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}
Resource (slots)	71	127	27	99	15	3	5	9	2	2
Mapping: (Width, Height)	3, 24	5, 26	1, 27	4, 25	3, 3	4, 1	5, 1	4, 3	4, 1	3, 1
Over allocation	1	3	0	1	0	1	0	3	2	1

First, the algorithm sorts all resource allocations in descending order of the area (Step 1). That results in A_2, A_4, A_1, A_3, A_5, A_8, A_7, A_6, A_9, and A_{10}, respectively. The DL subframe area mapping is done from right to left and bottom to top. The largest resource allocation $A_2=127$ is chosen first. Applying step 2 we get a width of $\lceil 127/30 \rceil = 5$ columns and a height of $\lceil 127/5 \rceil = 26$. The rectangle 5×26 results in an over allocation of just 3 slots.

Mapping of A_2 leaves a strip of 5×4. In step 3, the algorithm chooses the next largest resource allocation that can fit in to this space. It is $A_5 =15$. This is mapped as $5\times\lceil 15/5 \rceil$ or 5×3, resulting in no over allocation slot and leaves a space of 5×1 on top. A_7 can perfectly fit within this space with 5×1. Since there is no left-over space within this strip, we repeat step 2 by moving horizontally to the left. A_4 = 99, the next largest resource allocation, is mapped into the DL subframe in to a rectangle of width $\lceil 99/30 \rceil = 4$ and height $\lceil 99/4 \rceil = 25$. The rectangular mapping of 4×25 results in an over allocation of 1 and a left-over strip of 4×5 on the top of the mapping.

We move to step 3 to fill this 4×5 strip. At this time, A_8 = 9 being mapped to a rectangle of $\lceil 9/4 \rceil \times 4$ results in an over allocation of 3 and a left-over space of 4×2 on the top. Before we move to step 2, A_6 and A_9 are mapped and result in one and two over allocation slots respectively.

The next largest resource allocation, A_1, is mapped to 3×24 with one over allocation slot and a 3×5 left-over space. The only resource here, A_{10}, can fit within this space and results in one over allocation slot. At this time, although there is still 3×4 left-over space, A_3= 27, the only one unmapped resource allocation, can't fit in this space, and unfortunately the algorithm reaches the maximum frame width. As a result, the algorithm needs to use one additional column and then map A_3 as 1×27, and finally the algorithm terminates.

In this particular example, the total of over allocation slots is 1 + 3 + 1 + 1 + 3 + 2 + 1= 12, and the total of unused slots is 1 × 3 + 3 × 5 = 18 as shown by the dark and light shaded areas in Fig. 3 respectively. The efficiency of the algorithm (percentage of space used) is 92.30% with over allocation slots and unused slots being counted as wasted.

V. PERFORMANCE EVALUATION

In this section, we present numerical results comparing the eOCSA algorithm with the ideal (full-search) algorithm.

To assure the QoS requirements, we assume that the scheduler strictly allocates the resource allocation in each frame, and the total resource allocation slots are 360 slots. We also assume each MS needs one burst. The number of MSs is randomly chosen from 1 to 49. The resource allocation for each MS is also randomly generated in the range from 1 to 360 slots. The over allocations and unused slots are averaged and normalized over 100 trials.

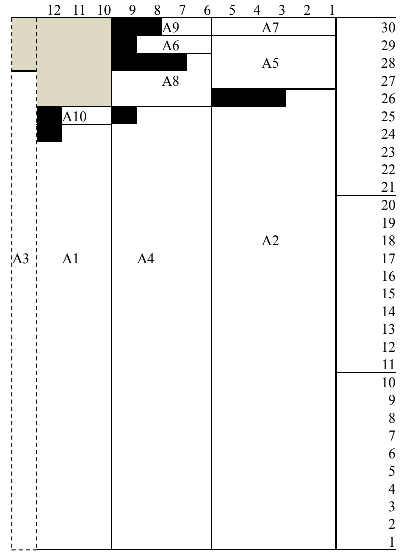

Fig. 3. Two-dimensional downlink burst mapping: Example

The results are shown in Fig. 4 in terms of the normalized over allocations and unused slots versus the number of MSs. On average, the normalized over allocation and unused slots are 0.0088 and 0.0614 compared to the ideal mapping. These normalized numbers includes the additional columns required to guarantee mapping of all resource allocations.

In addition to compare eOCSA with the ideal mapping, we also chose to compare eOCSA with the mapping algorithm by Takeo Ohseki *et al*. [5] and OCSA [3]. Each resource allocation is treated as a single burst. We could not compare eOCSA with other published algorithms for various reasons. For example, Yehuda Ben-Shimol *et al*. [7] provide no details of how to map the resources to unused spaces if their sizes are over multiple rows. Bacioccola *et al.* [6], assume that it is possible to have more than one burst per subscriber. This violates our goal of minimizing burst overhead. Our analysis shows eOCSA can support more than 30 MSs for the case where binary-tree full search supports only 8 subscribers [4].

Fig. 5 shows additional columns required for eOCSA, OCSA, and Takeo Ohseki's algorithm for 100 cases. On the average eOCSA only requires 0.93 (or 1) additional column. If the scheduler debits the over allocations and unused slots from future allocations, the number of additional columns in some frames can be reduced.

With the same configuration, Figs. 6 and 7 show the results of the algorithm by Takeo Ohseki *et al.* and the OCSA algorithm, again compared to the ideal mapping. On the average, the normalized unused slots and over allocation slots are 0.5198 and 0.0029 (Takeo Ohseki *et al*.), and 0.1309 and 0.0050 (OCSA), respectively. The average number of additional columns is 16.93 and 2.34 columns with these two algorithms. This behavior is because the unused slots are not considered.

VI. CONCLUSIONS

In this paper, we introduce a heuristic algorithm called eOCSA for two-dimensional downlink burst mapping for IEEE 802.16e Mobile WiMAX networks. The algorithm meets the rectangular allocation constraint and achieves high throughput by minimizing a left-over space, and optimizes the

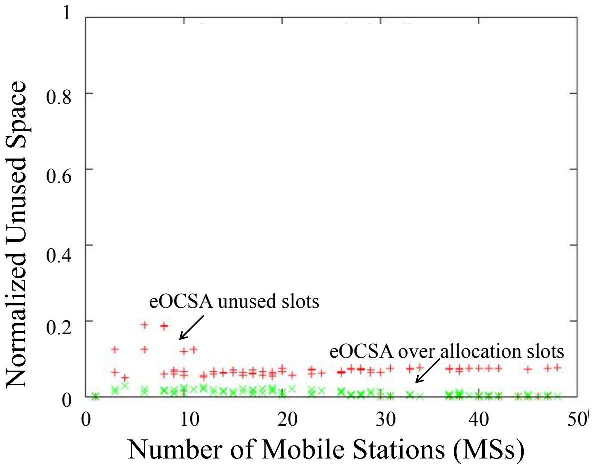

Fig. 4. Normalized unused space vs. number of MSs (eOCSA)

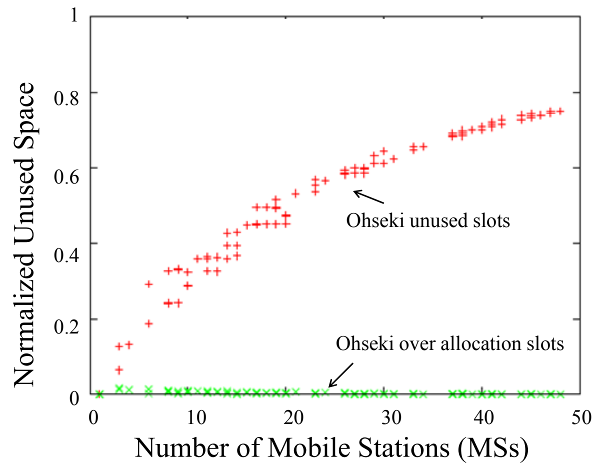

Fig. 6. Normalized unused space vs. number of MSs (Takeo Ohseki *et al.*)

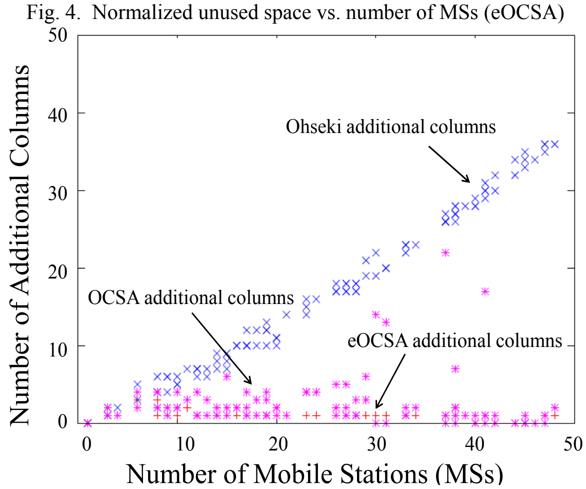

Fig. 5. Additional columns vs. number of MSs

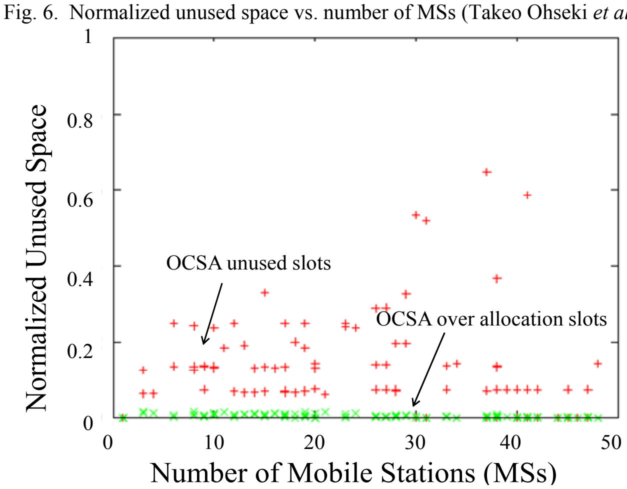

Fig. 7. Normalized unused space vs. number of MSs (OCSA)

energy consumption at mobile stations (MSs) by minimizing the receive time for the MS. To maximize the throughput, the algorithm considers the mapping in a descending order of the size of resource allocations. The mapping is done from right to left and bottom to top of the DL subframe. This allows space for variable portions of the DL-MAP and the UL-MAP to be adjusted accordingly in the left part of the subframe.

Since eOCSA is a heuristic algorithm, there is a tradeoff between the throughput optimality and the computation complexity. The throughput may be improved with more complex algorithms such as by making a recursion for both vertical and horizontal mapping.

We also compared the performance of eOCSA with ideal full-search algorithm and found that eOCSA provides 93% throughput compared to a full search algorithm. On average, the number of columns required is one column more than that required to accommodate the sum of all allocations.

REFERENCES

[1] IEEE P802.16Rev2/D2, "DRAFT Standard for Local and metropolitan area networks," Part 16: Air Interface for Broadband Wireless Access Systems, Dec. 2007, 2094 pp.

[2] WiMAX Forum, "WiMAX System Evaluation Methodology V2.1," Jul. 2008, 230 pp. URL=http://www.wimaxforum.org/technology/documents

[3] C. So-In, R. Jain, and A. Al-Tamimi, "OCSA: An algorithm for Burst Mapping in IEEE 802.16e Mobile WiMAX Networks," To appear *in the 15th Asia-Pacific Conference on Comm. (APCC 2009)*, Oct. 2009.

[4] C. Desset, E. B. de Lima Filho, and G. Lenoir, "WiMAX Downlink OFDMA Burst Placement for Optimized Receiver Duty-Cycling," in *Proc. IEEE Int. Conf. Comm.*, 1007, pp. 5149-5154.

[5] T. Ohseki, M. Morita, and T. Inoue, "Burst Construction and Packet Mapping Scheme for OFDMA Downlinks in IEEE 802.16 Systems," in in *Proc. IEEE Global Telecomunications Conf.*, 2007, pp. 4307-4311.

[6] A. Bacioccola, C. Cicconetti, L. Lenzini, E.A.M.E. Mingozzi, and A.A.E.A. Erta, "A downlink data region allocation algorithm for IEEE 802.16e OFDMA," in *Proc. Int. Conf. Information, Comm. and Signal Processing*, 2007, pp. 1-5.

[7] Y. Ben-Shimol, I. Kitroser, and Y. Dinitz, "Two-dimensional mapping for wireless OFDMA systems," in *IEEE Trans. Broadcasting*, vol. 52, no. 3, pp. 388-396, Sept. 2006.

[8] A. Lodi, S. Martello, and M. Monaci, "Two-dimensional packing problems: A survey," in *European Journal Operational Research*, vol. 141, pp. 241-252, Sept. 2002.

[9] F. Clautiaux, J. Carlier, and A. Moukrim, "A new exact method for the two-dimensional orthogonal packing problem," in *European Journal Operational Research*, vol. 127, pp. 1196-1121, Dec. 2007.

[10] E. Hopper and B.C.H. Turton, "A Review of the Application of Meta-Heuristic Algorithms to 2D Strip Packing Problems," in *Artif. Intell. Rev. Journal*, vol. 16, pp. 257-300, Dec. 2001.

[11] M-R. Garey and D-S. Johnson, "Computers and Intractability: A Guide to the Theory of NP-Completeness," W.H. Freeman, 340 pp., Jan. 1979.

[12] C. So-In, R. Jain, and A. Al-Tamimi, "Capacity Evaluation for IEEE 802.16e Mobile WiMAX," To appear in *Journal of Comp. Systems, Networks, and Comm. (Special Issue in WiMAX, LTE, and WiFi Internetworking)*, April. 2010.

[13] C. So-In, R. Jain, and A. Al-Tamimi, "Scheduling in IEEE 802.16e Mobile WiMAX Networks: Key Issues and a Survey," in *IEEE Journal on Selected Areas in Comm.,* vol. 27, no. 2, pp. 156-171, Feb. 2009