

# 國立交通大學

電信工程研究所

碩士論文

WiMAX 系統下兩階級數據的數據映射演算法之改進

An Enhanced Data Mapping Algorithm for Two-Level Requests  
in WiMAX Systems

研究生：黎廷勇

指導教授：李程輝 教授

中華民國一百零二年七月

WiMAX 系統下兩階級數據的數據映射演算法之改進

An Enhanced Data Mapping Algorithm for Two-Level Requests  
in WiMAX Systems

研究生：黎廷勇

Student : Dinh-Dung Le

指導教授：李程輝教授

Advisor : Prof. Tsern-Huei Lee



Submitted to Institute of Communications Engineering  
College of Electrical Engineering and Computer Engineering  
National Chiao Tung University  
in partial Fulfillment of the Requirements  
for the Degree of  
Master  
in  
Communications Engineering

July 2013

Hsinchu, Taiwan, Republic of China

中華民國一百零二年七月

# WiMAX 系統下兩階級數據的數據映射演算法之改進

研究生：黎廷勇

指導教授：李程輝 教授

## 國立交通大學 電信工程研究所

### 摘要

IEEE 802.16e 標準，採用 OFDMA PHY 一般被稱為行動 WiMAX (Mobile WiMAX)，在過去的幾年中已經引起了無線通信業界的關注。它是一個基於 IP 網路的高速遠距離無線寬帶接取技術的優化。在行動 WiMAX 系統的下行鏈路，數據都被映射到矩形區域。在矩形的限制下，最大化吞吐量是一個 NP 完全問題。過去有一些啟發式數據映射算法被提出，他們試圖實現系統的高吞吐量。其中有大多數的演算法只處理一種類型的數據。在最近的研究中，一個可映射兩級別的數據請求（即緊急和非緊急數據）的數據映射演算法被提出。然而，它在決定數據映射的順序時卻只有考慮數據請求所需的槽數量。

在這篇論文中，我們提出的增強版本，同時考慮每個行動用戶所需的槽數量和所使用的調變編碼機制。模擬結果顯示，與以前的設計相比，增強的演算法能為緊急數據提供更高的吞吐量。

**關鍵字：**IEEE 802.16e、數據映射、行動 WiMAX、兩階級數據。

# An Enhanced Data Mapping Algorithm for Two-Level Requests in WiMAX Systems

Student: Dinh-Dung Le

Advisor: Prof. Tsern-Huei Lee

Institute of Communications Engineering,  
National Chiao Tung University

## Abstract

The IEEE 802.16e standard, which adopts OFDMA PHY and is generally known as Mobile WiMAX, has grabbed the attention of wireless communication industry for the past few years. It is a technology optimized for IP-based high speed long-distance wireless broadband access. In the downlink of Mobile WiMAX systems, data requests have to be mapped into rectangular regions. Such a constraint results in an NP-complete problem in order to maximize throughput. Several heuristic data mapping algorithms were proposed trying to achieve high system throughput. Most of the algorithms handle only one type of data. In a previous research, a data mapping algorithm which maps two levels of requests, i.e., urgent and non-urgent data, has been presented. However, only the number of required slots were considered in determining the order of data mapping.

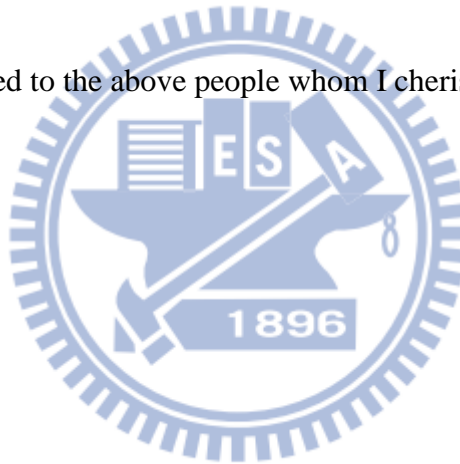
In this thesis, we propose an enhanced version which considers both the number of required slots and the modulation level for each mobile station. Simulation results show that, compared with previous design, the enhanced algorithm serves more urgent data with higher throughput.

**Keywords:** IEEE 802.16e; Data Mapping; Mobile WiMAX; Two-Level Requests.

# Acknowledgements

I would like to acknowledge my advisor, Prof. Tsern-Huei Lee, for his valuable guidance throughout my research. I would also like to express my appreciations to Prof. Yaw-Wen Kuo for his kind helps and useful suggestions. Next, I wish to thank all members in 823 Lab, especially Yang-Zi Yang and Tzu-Han Wu, for helping me through my academic years; they all played an important role during my studies here in Taiwan. Last but not least, I would like to thank my family for their encouragements and supports.

This thesis is dedicated to the above people whom I cherish forever and ever.



# Contents

摘要.....	i
Abstract.....	ii
Acknowledgements.....	iii
Contents.....	iv
List of Tables.....	vi
List of Figures.....	vii
Chapter 1 Introduction.....	1
1.1 An Overview of WiMAX System.....	1
1.2 Downlink Burst Mapping Problem.....	3
1.3 Objective.....	5
1.4 Thesis Organization.....	6
Chapter 2 Related Works.....	7
2.1 Enhanced OCSA (eOCSA).....	7
2.2 Two-Level Requests Mapping (TLRM).....	9
2.3 DL Data Mapping Examples using Related Algorithms.....	10
2.3.1 DL data mapping using eOCSA.....	11
2.3.2 DL data mapping using TLRM.....	14
Chapter 3 Proposed Algorithm.....	18
3.1 System Model and Definition.....	18
3.2 E-TLRM Description.....	19
3.2.1 E-TLRM first phase.....	19
3.2.2 E-TLRM second phase.....	21
3.3 An E-TLRM Example.....	23

Chapter 4	Performance Evaluation .....	27
Chapter 5	Conclusions .....	33
Bibliography.....		34



## List of Tables

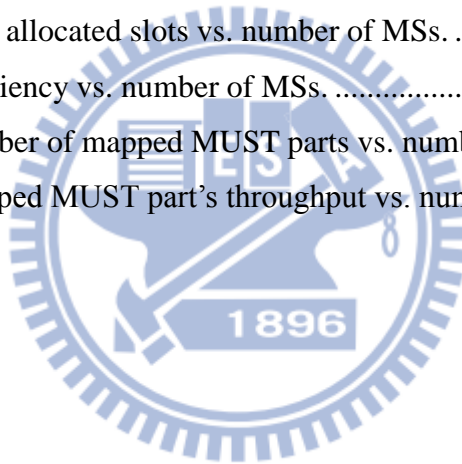
Table 2.1	The set of data requests using in related algorithm examples.....	10
Table 2.2	eOCSA example results.....	13
Table 2.3	TLRM example results.....	16
Table 3.1	An example of proposed algorithm.....	23
Table 3.2	E-TLRM example results.....	26
Table 4.1	System simulation parameters.....	28





## List of Figures

Fig. 1.1	A simple OFDMA DL sub-frame structure. ....	2
Fig. 1.2	Resource wastage in DL Mapping. ....	5
Fig. 2.1	An example of mapping DL bursts using eOCSA. ....	8
Fig. 2.2	DL data mapping using eOCSA. ....	13
Fig. 2.3	The first phase of TLRM algorithm. ....	15
Fig. 2.4	The second phase of TLRM algorithm. ....	17
Fig. 3.1	Example of a general burst in DL sub-frame. ....	21
Fig. 3.2	An Example of DL data mapping using E-TLRM algorithm. ....	25
Fig. 4.1	Average unused slots vs. number of MSs. ....	29
Fig. 4.2	Average over allocated slots vs. number of MSs. ....	30
Fig. 4.3	Average efficiency vs. number of MSs. ....	31
Fig. 4.4	Average number of mapped MUST parts vs. number of MSs. ....	31
Fig. 4.5	Average mapped MUST part's throughput vs. number of MSs. ....	32



# Chapter 1 Introduction

## 1.1 An Overview of WiMAX System

Mobile WiMAX based on the IEEE 802.16e standard [1] has been introduced as the first broadband wireless access technology that supports both fixed and mobile environments. The IEEE 802.16e standard specifies the physical (PHY) and Medium Access Control (MAC) layers while the WiMAX Forum ensures interoperability between different manufactures' devices. In fact, there are three different PHY layers defined: Single-Carrier PHY, Orthogonal Frequency-Division Multiplexing (OFDM) PHY, and Orthogonal Frequency-Division Multiple Access (OFDMA) PHY. Compared to the other PHY technologies, the OFDMA PHY is superior and has therefore been selected by the WiMAX Forum because of its features such as reducing the granularity in the radio resource mechanism, using the available power more efficiently, and leading to a significant cell range extension in the uplink transmission [2].

In the IEEE 802.16e OFDMA system, while both Frequency Division Duplexing (FDD) and Time Division Duplexing (TDD) operation modes are specified, there are many initial products and deployment scenarios currently focusing on TDD operation because of the flexibility to partition downlink (DL) and uplink (UL) resources and better channel reciprocity to support closed enhancing techniques [3]. In the TDD system, the time domain is divided into 5ms frames, which are separated into DL and UL sub-frames [4].

As can be seen in Fig. 1.1, an OFDMA DL sub-frame begins with a DL preamble which is used for frame synchronization, channel state estimation, received signal

strength, and signal-to-interference-plus-noise ratio (SINR) estimation; and followed by Frame Control Header (FCH), downlink map (DL-MAP) and uplink map (UL-MAP) messages that describe the structure and composition of frame.

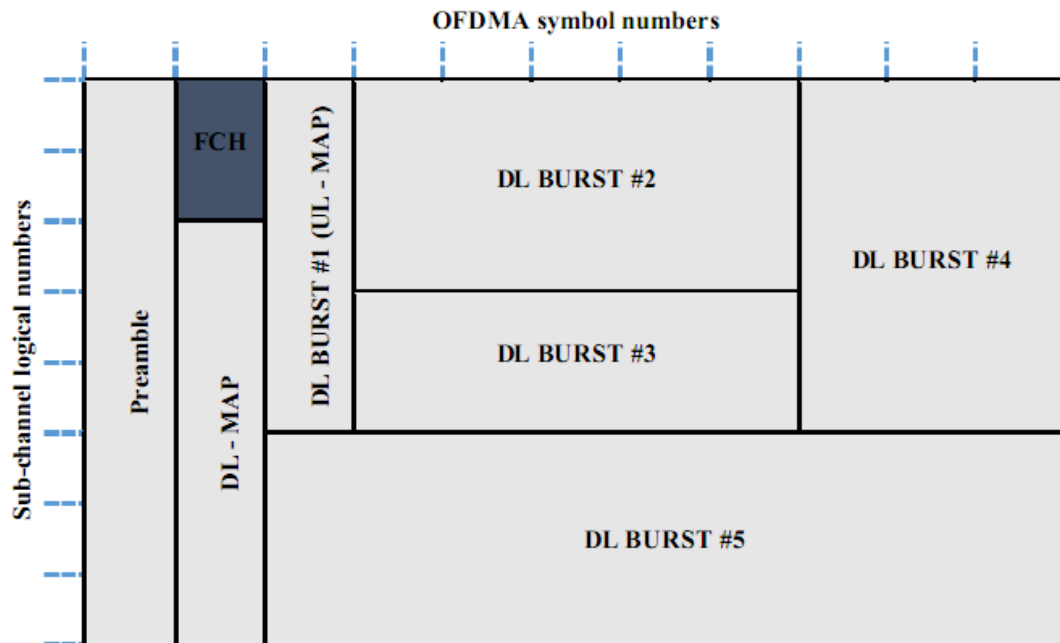


Fig. 1.1 A simple OFDMA DL sub-frame structure.

According to the IEEE 802.16e specification, the units of resource allocation are slots, bursts and permutation zones. A slot is the minimum possible data allocation unit in time and frequency dimensions. It's a combination of a sub-channel and one or more OFDMA symbols depending on sub-carrier permutation mode. On the downlink, a burst denotes a rectangular region which consists of a group of contiguous logical sub-channels in a group of contiguous OFDMA symbols. Each burst is transmitted using a single Modulation and Coding Scheme (MCS) and may include MAC protocol data units (PDUs) intended for one or more users. A permutation zone denotes an allocation in time when one particular subcarrier permutation mode is used to map data symbols onto sub-channels. A downlink sub-frame may contain more than one

permutation zone.

In this thesis, we only consider the downlink sub-frame with Partially Used Sub-Channelization (PUSC) zone wherein the sub-carriers are fairly distributed over the entire frequency band by using PUSC permutation mode. With PUSC mode, a slot will be a combination of one sub-channel and two OFDMA symbols; according to WiMAX Forum specified parameters, with 10 Mhz spectrum, there are 30 sub-channels, each one consists of 28 sub-carriers.

The IEEE 802.16e standard has defined 5 scheduling service classes with different QoS requirements, including bandwidth, packet loss, delay, and delay jitter: Unsolicited Grand Service (UGS), real-time Polling Service (rtPS), extended real-time Polling Service (ertPS), non-real-time Polling Service (nrtPS), and Best Effort (BE) [5][6]; each class has different QoS parameters.

Moreover, the standard also specifies that each data burst has to be mapped into a rectangular region of the downlink sub-frame. This constraint makes the DL mapping as two-dimensional downlink burst mapping problem introduced in the next section.

## 1.2 Downlink Burst Mapping Problem

The two-dimensional downlink burst mapping problem in Mobile WiMAX system is a variation of the bin packing problem, which is known to be NP-complete [7].

In order to define the problem, assume that the DL sub-frame is a two-dimensional matrix of slots, and the area of each burst is expressed in term of slots. Let  $c$  and  $s$  be the number of sub-channels and number of time slots in a DL sub-frame, respectively. Thus the total resource in a frame is  $B = s \times c$  slots. Given a set of  $n$  items

$\{b_1, b_2, \dots, b_n\}$ , and each item  $b_i$  has a size  $A_i$ ,  $1 \leq i \leq n$ . All items are mapped into  $B$  under the following constraints:

- No overlap between any two rectangular bursts regions.
- $A_i \leq W_i \times H_i$ . Where  $W_i$  and  $H_i$  are the width and the height of the rectangle burst assigned to item  $i^{\text{th}}$ , respectively.
- $W_i \leq s$  and  $H_i \leq c$  for all  $i$ .

However, the OFDMA downlink burst mapping is different from the typical bin packing. First, the dimensions of rectangles are predetermined in the bin packing, while the bursts can be shaped flexibly. Second, there can be *resource wastage* due to the size mismatch between the data requests and the allocated burst, which does not exist in the bin packing. When the actual data belonging to a request cannot fill the whole rectangular region of a burst, the vacant slots are considered as *over allocated slot* wastage within the burst. Besides, in some cases the remaining slots in the DL sub-frame cannot form a rectangle to fit any unmapped requests. These remaining slots are called *unused slot* wastage outside the bursts. Both these wasted slots reduce the efficiency and should be minimized. Fig. 1.2 is an example about resource wastage in DL Mapping.

For this example, over allocated slots and unused slots are shown in black and white, respectively.

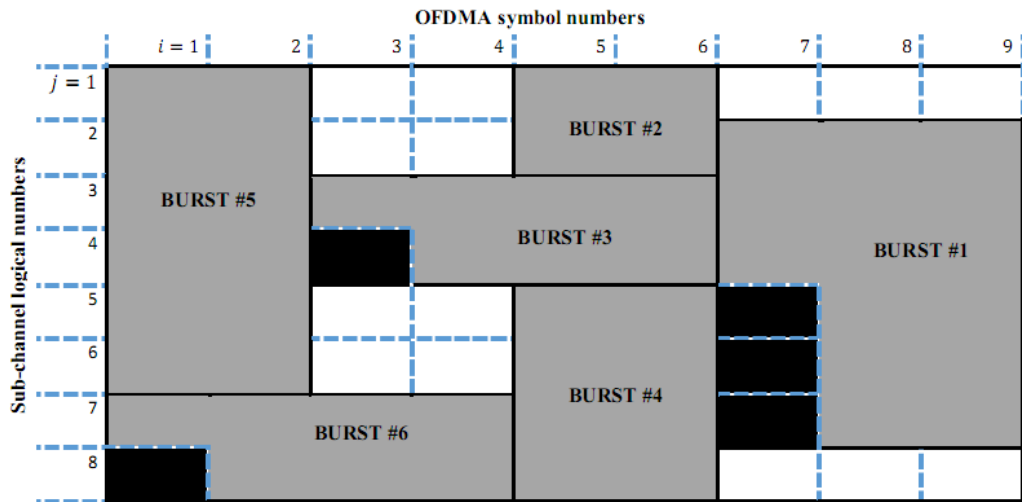


Fig. 1.2 Resource wastage in DL Mapping.

### 1.3 Objective

The problem of efficiently mapping the DL data requests into rectangular regions in the DL sub-frame as mentioned above is not addressed by the IEEE 802.16e standard, and is left as an implementation issue. As a result, several DL mapping algorithms have been proposed in literature recently [8]-[11]. In particular, a mapping algorithm which handles two levels of data requests, i.e., urgent and non-urgent data, was presented in [12]. For convenience, we shall refer to this algorithm Two-Level Requests Mapping (TLRM).

In TLRM algorithm, each user's request can consist of urgent and non-urgent data. The algorithm consists of two phases. Phase 1 maps all requests with both urgent and non-urgent parts into the DL sub-frame, and Phase 2 returns some mapped non-urgent data parts so that more urgent data parts can then be mapped. The goal of TLRM algorithm is to map the real-time traffic effectively but do not focus on the urgent part in the first phase. Moreover, if several MCSs are used, the mapping order based on the required number of slots becomes inefficiency. Suppose that there are two requests with

the same size. The one with a worse MCS is mapped prior to the other which may be unmapped when there are no enough slots for both, resulting in a low spectrum efficiency.

The purpose of this thesis is to present an enhanced version of the TLRM algorithm. In addition to the number of required slots, we also consider achievable Modulation and Coding Scheme (MCS) of urgent data in determining the order of data mapping. Simulation results show that, compared with previous design, the enhanced algorithm serves more urgent data with higher throughput. Simulation results show that, compared with previous design, the enhanced algorithm serves more urgent data with higher throughput.

## **1.4 Thesis Organization**

The rest of this thesis is organized as follows. In Chapter 2, the related algorithms eOCSA and TLRM will be reviewed in detail. Our proposed data mapping algorithm, called Enhanced Two-Level Requests Mapping (E-TLRM) in WiMAX Systems, is briefly introduced and described in Chapter 3. Details of the performance evaluation of our E-TLRM algorithm when compared with eOCSA and TLRM are given in Chapter 4. Finally, Chapter 5 gives our conclusions.

# Chapter 2 Related Works

This chapter presents a comprehensive introduction to related DL mapping algorithms, including the eOCSA and TLRM algorithms. eOCSA [14] is the enhanced version of One Column Stripping with non-increasing Area first mapping (OCSA) [13], aims to keep the mapping operational complexity low. Based on the eOCSA algorithm, TLRM is introduced to effectively map the sensitive real-time traffic.

## 2.1 Enhanced OCSA (eOCSA)

In eOCSA, the data requests are sorted in the descending order (largest first) and are mapped into the DL sub-frame from bottom to top and from right to left to allow the space for variable portions of the DL-MAP.

Let  $H$  denote the maximum height in the DL sub-frame and  $\mathbf{A} = \{A_i\}_{i=1}^N$  denote the set of  $N$  data requests. The eOCSA scheme consists of four steps. In the **first step**, the algorithm sorts the set of data requests in the descending order and selects the largest request, say,  $A_k$ . In the **second step**, also known as *vertical mapping*, eOCSA maps the request  $A_k$  to a rectangle with width  $W_k$  and height  $H_k$ .

$$W_k = \lceil A_k / H \rceil \quad (1)$$

$$H_k = \lceil A_k / W_k \rceil \quad (2)$$

Where,  $\lceil \cdot \rceil$  denotes the *ceiling function*, which ensures that the mapped region is bigger than required data request ( $W_k \times H_k \geq A_k$ ) and has the minimum possible width (minimizing MS active time and energy).



The remaining unallocated space is handled in the *third step*, during which the *horizontal mapping* takes place as the algorithm looks for the next largest data request that will fit into the space left on the top of the burst mapped in the previous step. But here, the widths of all following bursts mapped in the strip are fixed to that of the burst mapped in the previous step. In other words, the height of each burst allocated in the strip is calculated as the *ceiling* function of the data request divided by the fixed width. This process is repeated until no space can be allocated or there is no data request that can fit in this space. Next, the algorithm moves leftward to fill the remaining empty columns in the DL sub-frame and repeats the process from the second step.

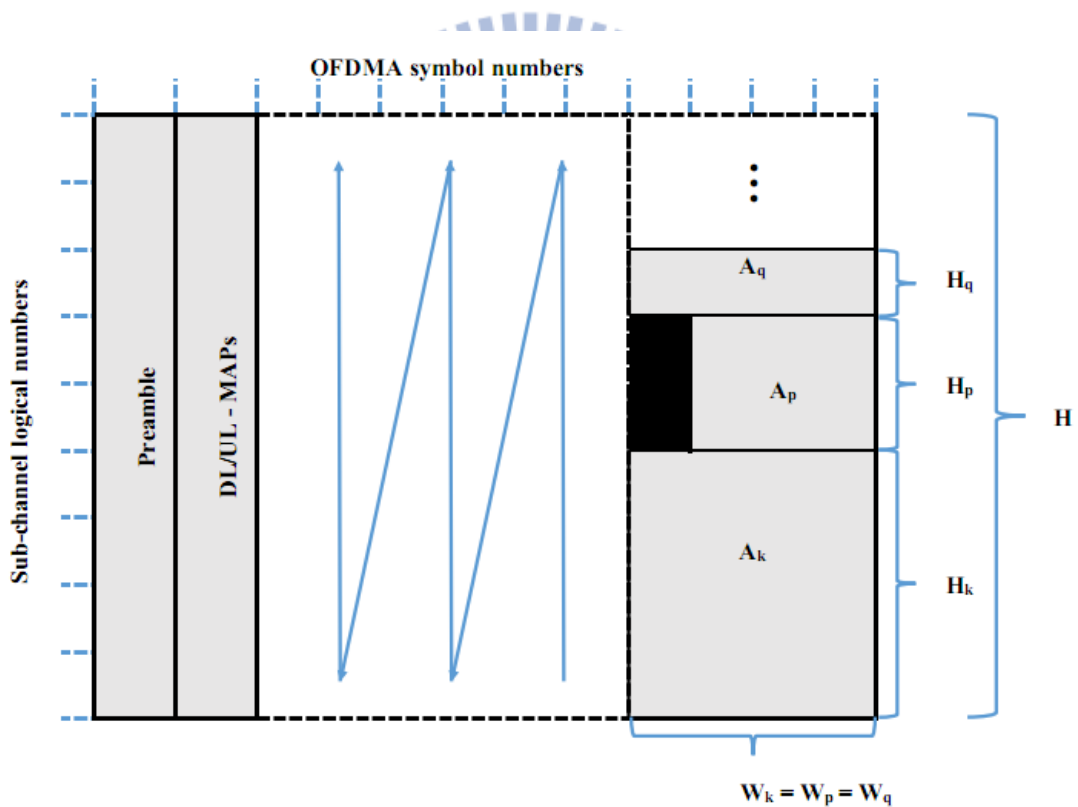


Fig. 2.1 An example of mapping DL bursts using eOCSA.

Fig. 2.1 shows the process of moving vertically and horizontally from bottom to top and right to left. Note that when the data request is smaller than the mapped region, there are some over allocated slots inside the burst (presented by black in this example).

## 2.2 Two-Level Requests Mapping (TLRM)

In fact, a data request can consist of two parts (or two levels): MUST part and WISH part. The MUST part is the urgent data which will be dropped if it is not mapped and the WISH part represents non-urgent data that can be transmitted in later frames. For instance, rtPS, UGS, and ertVR regarded as urgent data called MUST data; while nrtPS and BE are treated as non-urgent data called WISH data.

TLRM algorithm is divided into two phases. In the first phase, TLRM basically uses a modified version of eOCSA algorithm, with two-level requests. However, for two-level requests, the scheme requires that when mapping the data requests into rectangles, the MUST part is allocated first, followed by the WISH part. The first phase can be separated into 4 steps. In the first step, the algorithm selects a strip which is a group of contiguous columns in the DL sub-frame. The strip has the width  $W$  and an largest empty space called maximum available rectangle space, say  $R_{i,j}$ . In the second step, the TLRM selects the largest two-level request  $A_k$  in the set of data requests such that  $A_k$  can be mapped into the  $R_{i,j}$  found in the previous step. The first phase terminates if no such  $A_k$  exists. In the third step, the algorithm maps  $A_k$  into  $R_{i,j}$ . If there is no mapped request in the strip, the algorithm maps  $A_k$  to a rectangle with width  $W^*$  and height  $H^*$ , where  $W^* = \lceil A_k/H \rceil$  and  $H^* = \lceil A_k/W \rceil$ . Here, if  $W^*$  is smaller than  $W$ , the algorithm divides the selected strip into 2 smaller strips of widths  $W^*$  and  $W - W^*$ . In case some requests had been mapped in the selected strip, the algorithm maps  $A_k$  to a rectangle with width  $W$  and height  $H^*$ , where  $H^* = \lceil A_k/W \rceil$ . Next, the algorithm updates the maximum available rectangle spaces in the strips and removes  $A_k$  from the set of data requests.

After the first phase is completed, usually there are some remaining requests that were not mapped. Since these “unmapped requests” might contain the MUST part and WISH part, the second phase consist of mapping as much MUST part as possible for those unmapped requests. Firstly, the algorithm selects the unmapped request  $A_k$  whose MUST part  $M_k$  is the largest one. Secondly, it selects a proper strip to map  $M_k$ . If  $M_k$  is less than the maximum available rectangle space  $R_{i,j}$  for some strips, the strip with the smallest  $R_{i,j}$  will be selected. Otherwise, the algorithm selects strip with largest over allocated slots and removes some WISH data of the mapped requests, so that  $M_k$  can fit in the strip. Thirdly,  $M_k$  is mapped into the selected strip same as the third step of the first phase. If there are unused slots in the strip, the scheme continues to map WISH part of  $A_k$ . Finally,  $A_k$  is removed from the set of data requests.

### 2.3 DL Data Mapping Examples using Related Algorithms

In this section, we provide 2 examples that help explain the operation of eOCSA and TLRM algorithms. Assume that there are 12 timeslots and 30 sub-channels in the DL sub-frame. There are 10 MSs with the data request set  $A$  as show in **Table 2.1**. The sum of all data requests is 360, in which the sum of the MUST part of all requests is equal to 180.

**Table 2.1 The set of data requests using in related algorithm examples**

Resource	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$	$A_7$	$A_8$	$A_9$	$A_{10}$
(slots)	38	8	86	32	41	3	115	7	17	13
MUST part	14	6	48	22	15	2	50	7	11	5
WISH part	24	2	38	10	26	1	65	0	6	8

### 2.3.1 DL data mapping using eOCSA

**First**, eOCSA sorts all the data requests shown in **Table 2.1** in the descending order of the request size and selects the largest request to map first (step 1). After finished step 1, the order of set  $A$  is  $\{A_7; A_3; A_5; A_1; A_4; A_9; A_{10}; A_2; A_8; A_6\}$  and the largest data request  $A_7 = 115$  is chosen to map first.

**Applying step 2**, we get a width of  $\lceil 115/30 \rceil = 4$  columns and a height of  $\lceil 115/4 \rceil = 29$  rows. The rectangle  $4 \times 29$  results in an over allocated of 1 slot. The downlink sub-frame mapping is done from right to left and from bottom to top. The mapping of  $A_7$  into the sub-frame leaves a strip of  $4 \times 1$ .

**In step 3**, the algorithm chooses the next largest data request that can fit into the remaining strip, which is  $A_6 = 3$ . And it is mapped as  $4 \times \lceil 3/4 \rceil$  or  $4 \times 1$ , resulting in 1 over allocated slot. Since there is no left-over space within this strip, we repeat step 2 by moving horizontally to the left.

The next largest data request is  $A_3 = 86$ , mapped into the DL sub-frame in a rectangle of width  $\lceil 85/30 \rceil = 3$  and height  $\lceil 85/3 \rceil = 29$ . The rectangle mapping of  $3 \times 29$  results in an over allocated of 1 slot and a left-over strip of  $3 \times 1$  on the top. Since there is no data request that can be fitted into the remaining space, eOCSA repeats step 2 by moving horizontally to the left.

The next largest data request is  $A_5 = 41$ , mapped into the DL sub-frame in a rectangle of width  $\lceil 41/30 \rceil = 2$  and height  $\lceil 41/2 \rceil = 21$ . The rectangle mapping of

$2 \times 21$  results in an over allocated of 1 slot and a left-over strip of  $2 \times 9$  on the top. The algorithm then moves to step 3 to fill the  $2 \times 9$  strip. The next largest data request that can fit into this strip is  $A_9 = 17$ , which is mapped to a rectangle  $2 \times \lceil 17/2 \rceil$  or  $2 \times 9$ , resulting in 1 over allocated slot and no left-over space on the top. Since there is no left-over space within this strip, we repeat step 2 by moving horizontally to the left.

At this time, the largest request in set  $A$  is  $A_1 = 38$ , which is mapped to a rectangle of  $2 \times 19$ , with no over allocated slot and a  $2 \times 11$  left-over space. The scheme moves again to step 3 to look for the largest request that can fit into the remaining space  $2 \times 11$ .  $A_{10} = 13$  being mapped to a rectangle of  $2 \times \lceil 13/2 \rceil$  or  $2 \times 7$ ; resulting in 1 over allocated slot and a left-over space of  $2 \times 4$  on the top. Then  $A_2 = 8$  being mapped to a rectangle of  $2 \times \lceil 8/2 \rceil$  or  $2 \times 4$ ; resulting in no over allocated slot and no left-over space of on the top. eOCSA continues to repeat step 2 by moving horizontally to the left.

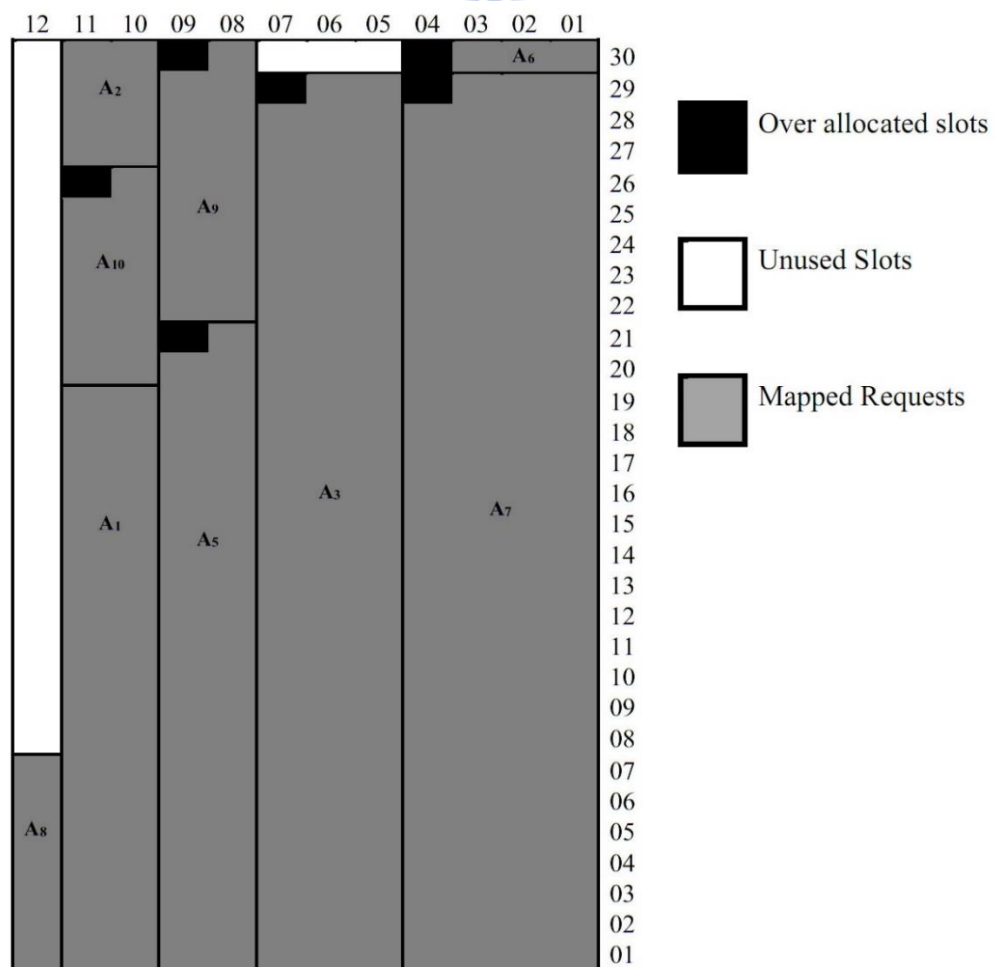
The next largest request  $A_4 = 32$  cannot fit in the remaining space of DL sub-frame, so the request  $A_8 = 7$  being mapped to a rectangle of  $1 \times \lceil 7/1 \rceil$  or  $1 \times 7$ ; with no over allocated and a  $1 \times 23$  left-over on the top. Unfortunately, there are no requests that can fit into this left-over space, so the algorithm is terminated.

In this example, the over allocated slots are  $1+1+1+1+1+1=6$ , and the total unused slots is  $3 \times 1 + 1 \times 23 = 26$ ; the efficiency of the algorithm (percented of used space) is 91.11%, with over allocated and unused slots being counted as wasted space. The final results of this example are shown as **Table 2.2**.

**Table 2.2 eOCSA example results**

<b>Over allocated Slots</b>	<b>6</b>
<b>Unused Slots</b>	<b>26</b>
<b>Unmapped data requests</b>	<b>1</b>
<b>Efficiency</b>	<b>91.11%</b>

The DL data mapping results of this example is shown in Fig. 2.2. Where the dark represent the over allocated slots, and the white represent the unused slots; areas in gray are all the mapped data requests.



**Fig. 2.2 DL data mapping using eOCSA.**

### 2.3.2 DL data mapping using TLRM

In this example, assume that each data request consists of two parts: MUST part and WISH part, which are shown as **Table 2.1**.

❖ **First Phase:**

This phase tries to map both MUST and WISH parts of a data request into the DL sub-frame.

In the first iteration, TLRM selects strip  $K_{1,12}$ , which has the width  $W_{1,12} = 12$  and the maximum available rectangle  $12 \times 30$ , and selects the largest request smaller than  $12 \times 30 = 360$  is  $A_7 = 115$  (step 1, 2). According to step 3, we get a width of  $\lceil 115/30 \rceil = 4$  columns and a height of  $\lceil 115/4 \rceil = 29$  rows. The data request  $A_7$  is mapped into the rectangle  $4 \times 29$  with 1 over allocated slot. The downlink sub-frame mapping is done from right to left and from bottom to top. Since the width of rectangle  $4 \times 29$  is smaller than the width  $W_{1,12}$ , strip  $K_{1,12}$  is split into  $K_{1,4}$  and  $K_{5,12}$ . The widths and maximum available rectangles of  $K_{1,4}$  and  $K_{5,12}$  are  $W_{1,4} = 4, 4 \times 1$  and  $W_{5,12} = 8, 8 \times 30$ , respectively.

In the second iteration, the maximum available rectangle  $8 \times 30$  of  $K_{5,12}$  and the next largest data request  $A_3 = 86$  are selected. According to step 3, we get a width of  $\lceil 86/30 \rceil = 3$  columns and a height of  $\lceil 86/3 \rceil = 29$  rows. The data request  $A_7$  is mapped into the rectangle  $3 \times 29$  with 1 over allocated slot. The downlink sub-frame mapping is done from right to left and from bottom to top. Since the width of rectangle  $3 \times 29$  is smaller than the width  $W_{5,12}$ , strip  $K_{5,12}$  is split into  $K_{5,7}$  and  $K_{8,12}$ . The widths

and maximum available rectangles of  $K_{5,7}$  and  $K_{8,12}$  are  $W_{5,7} = 3,3 \times 1$  and  $W_{8,12} = 5, 5 \times 30$ , respectively.

Similarly, the next largest data request is  $A_5 = 41$ , mapped into the DL sub-frame in a rectangle of width  $\lceil 41/30 \rceil = 2$  and height  $\lceil 41/2 \rceil = 21$  in the third iteration. The rectangle mapping of  $2 \times 21$  results in an over allocated of 1 slot and strip  $K_{8,12}$  is split into  $K_{8,9}$  and  $K_{10,12}$ .

The first phase is executed for six more iterations to sequentially map the data requests  $A_1, A_9, A_{10}, A_2, A_8$  and  $A_6$ . Fig. 2.3 shows the DL data mapping results of the first phase. Where the unused slots, over allocated slots, MUST parts, and WISH parts are presented in white, black, yellow, and blue, respectively.

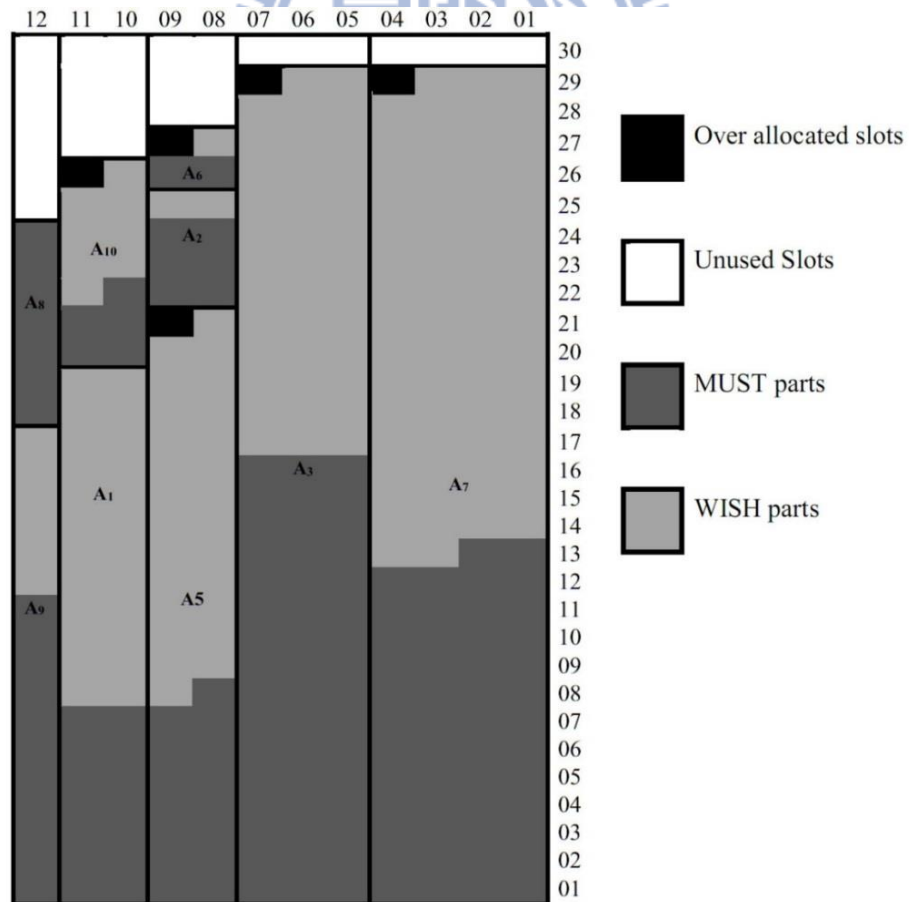


Fig. 2.3 The first phase of TLRM algorithm.



❖ **Second Phase:**

After finishing the first phase, we have 5 strips in the DL sub-frame:  $K_{1,4}$ ,  $K_{5,7}$ ,  $K_{8,9}$ ,  $K_{10,11}$ ,  $K_{12,12}$  and its maximum available rectangles are  $4 \times 1 = 4$ ,  $3 \times 1 = 3$ ,  $2 \times 3 = 6$ ,  $2 \times 4 = 8$ ,  $1 \times 6 = 6$ , respectively. The algorithm will try to map the MUST part  $M_4 = 22$  of the unmapped request  $A_4 = 32$  in the second phase.

According to step 2, TLRM chooses the strip  $K_{8,9}$  with the largest value of over allocated slots (2) to fit  $M_4$ . Since the number of unused slots in  $K_{8,9}$  is 6 slots, the algorithm will remove 14 slots belonging to the WISH parts of  $A_6$ ,  $A_2$ , and  $A_5$ . Then,  $M_4$  being mapped to the rectangle of  $2 \times 11$  in the top of  $K_{8,9}$ . Finally, the algorithm is terminated. The DL data mapping results of the second phase is shown in Fig. 2.4.

In this example, the over allocated slots are  $1+1+1=3$ , and the total unused slots is  $4 \times 1 + 3 \times 1 + 2 \times 4 + 1 \times 6 = 21$ ; the efficiency of the algorithm (percentage of used space) is 94.2%, with over allocated and unused slots being counted as wasted space. The final results of this example are shown as **Table 2.3**.

**Table 2.3 TLRM example results**

<b>Over allocated Slots</b>	<b>3</b>
<b>Unused Slots</b>	<b>21</b>
<b>Unmapped data requests</b>	<b>0</b>
<b>Efficiency</b>	<b>94.2%</b>

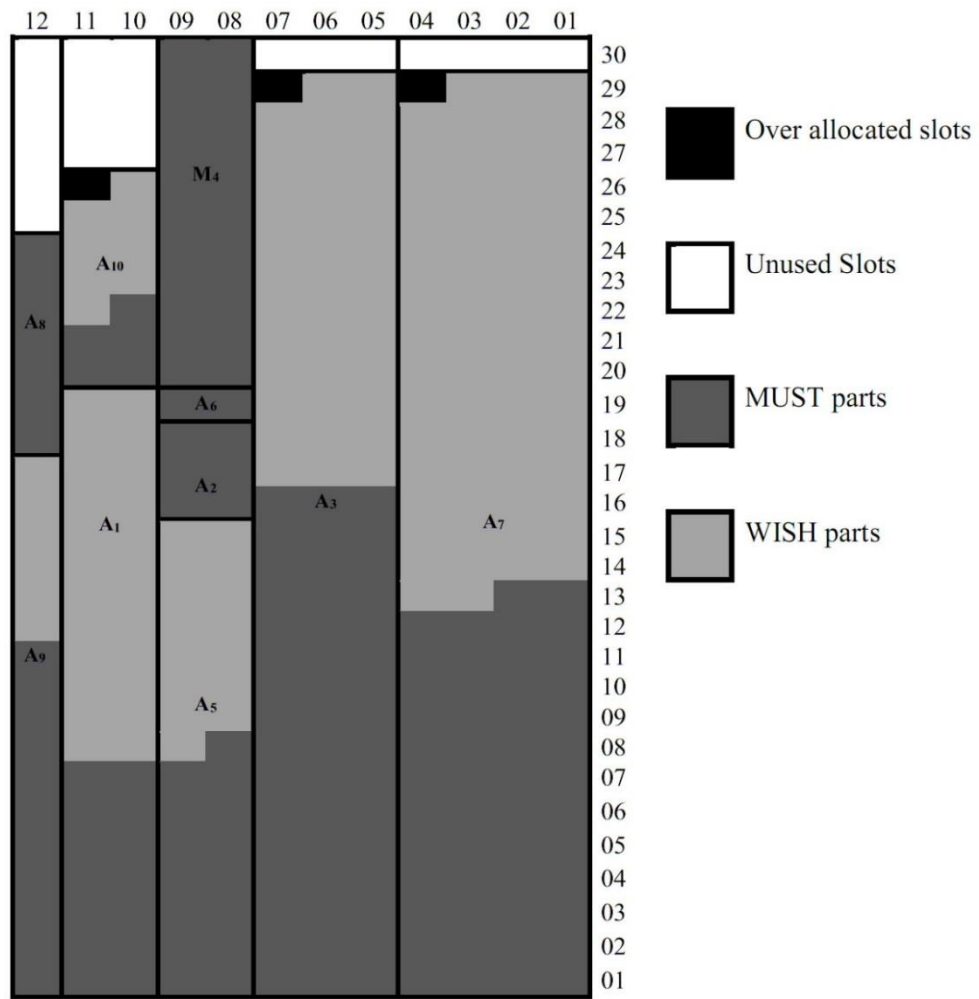


Fig. 2.4 The second phase of TLRM algorithm.

# Chapter 3 Proposed Algorithm

In general, Mobile WiMAX supports a variety of modulation and coding schemes (MCSs) to adapt and adjust the transmission based on channel conditions. As a result, a data request may require different burst sizes if different MCSs are used. In order to meet QoS requirements and maximize system throughput, larger urgent data with higher MCS should have higher mapping priority. This chapter describes our data mapping algorithm called Enhanced Two-Level Requests Mapping (E-TLRM), which considers both the size and available MCS of urgent data.

## 3.1 System Model and Definition

In this thesis, we adopt a WiMAX system which consists of a Base Station (BS) and  $N$  Mobile Stations (MSs). Each MS has a data request for the DL transmission. Let  $\Phi = \{1, 2, \dots, N\}$  represent the set of MSs where  $N$  denotes the number of MSs. Let  $r_i$  denote the data request of user  $i$  which consists of urgent data,  $r_i^M$  and non-urgent data,  $r_i^W$ , i.e.,  $r_i = r_i^M + r_i^W$ ,  $1 \leq i \leq N$ .

We only consider the downlink sub-frame with Partially Used Sub-Channelization (PUSC) zone wherein the sub-carriers are fairly distributed over the entire frequency band by using PUSC permutation mode. With PUSC mode, a slot will be a combination of one sub-channel and two OFDMA symbols.

In this system, several MCSs can be used to adapt the transmission rate to the channel conditions. Assume that the set of MCSs from the lowest to highest which are used in the Mobile WiMAX system will be mapped into the set of corresponding MCS

level from 1 to  $L$ . Based on the link quality, the BS decides a MCS level for each data request. Let  $m_i$  denote the MCS level that BS uses for the data request  $r_i$ ,  $1 \leq m_i \leq L$  and  $\mu_{m_i}$  is the number of bytes which a slot can carry with the MCS level  $m_i$ .

Let  $A$  denote the set of requests for mapping  $A = \{A_i\}_i^N$  (in number of slots). A request  $A_i$  will be the sum of two parts: the MUST part,  $M_i$ , and the WISH part,  $W_i$ , i.e.,  $A_i = M_i + W_i$ , where  $M_i = \lceil r_i^M / \mu_{m_i} \rceil$  and  $W_i = \lceil r_i^W / \mu_{m_i} \rceil$ ,  $1 \leq i \leq N$ .

Assume that there are  $s$  timeslots and  $c$  sub-channels in the DL sub-frame. Let  $\Omega = \{(x, y) | 1 \leq x \leq s, 1 \leq y \leq c\}$  be the set of all slots in the sub-frame. Let  $K_{i,j}$  be a vertical strip which consists of columns  $i$  to  $j$ ,  $1 \leq i \leq j \leq s$ . A rectangle  $R$  is denoted by  $[(x_0, y_0); w, h]$ , where  $(x_0, y_0)$ ,  $w$ , and  $h$  are its bottom-right corner, the width, and the height, respectively. Also, denote  $|R|$  as the area of  $R$ . Let the rectangle  $R_{i,j}$  be the maximum available rectangle in  $K_{i,j}$ .

## 3.2 E-TLRM Description

E-TLRM, similar to TLRM, is divided into two phases. **The first phase** consists of mapping all two-level requests with both MUST parts and WISH parts in descending order of MCS level and MUST part's size into the DL sub-frame. **The second phase** consists of returning some mapped requests' WISH parts, so that more MUST parts of unmapped requests can be mapped.

### 3.2.1 E-TLRM first phase

The first phase consists of four steps. The proposed algorithm maps slots to a request iteratively. Let  $\Psi$  be the set of all strips, and initially  $\Psi = \{K_{1,s}\}$ .

- **Step 1:** Sort the set of requests  $\mathbf{A}$  in the descending order of MCS level  $m_i$ . Two requests using the same MCS level are sorted according to MUST part's size  $M_i$ . Requests with the same MCS level and MUST part's size are sorted based on the sizes of their WISH parts. Ties are broken arbitrarily. For convenience, the set of sorted requests is called sorted\_ $\mathbf{A}$  whose elements are mapped sequentially. Consequently, requests with higher MCS levels have higher priority. The first phase terminates after the last element in sorted\_ $\mathbf{A}$  is handled.

- **Step 2:** Let  $A_k$  be the request to be mapped. Select the first strip in  $\Psi$ , say  $K_{i^*,j^*}$ , which can accommodate  $A_k$  in its maximum available rectangle  $R_{i^*,j^*}$ .  $K_{i^*,j^*}$  and  $R_{i^*,j^*}$  are denoted by  $[(i^*,1);(j^*-i^*+1),c]$  and  $[(i^*,y');(j^*-i^*+1),h']$ , respectively. In case that no such strip exists, we proceed to next request with the same operation.

- **Step 3:** Map  $A_k$  to  $R_{i^*,j^*}$ . There are two cases for  $K_{i^*,j^*}$ :

i) If  $|K_{i^*,j^*}| = |R_{i^*,j^*}|$ , there is no requests mapped in  $K_{i^*,j^*}$  and  $K_{i^*,j^*}$  must be

$K_{i^*,s}$ .

a. Calculate  $w = \lceil A_k / c \rceil$  and  $h = \lceil A_k / w \rceil$ . Allocate the rectangle

$$R = [(i^*,1);w,h] \text{ for request } A_k.$$

b. If  $w < s - i^* + 1$ , then split the strip  $K_{i^*,j^*}$  into  $K_{i^*,i^*+w-1}$  and  $K_{i^*+w,s}$ .

$$\text{Update } \Psi = (\Psi - \{K_{i^*,s}\}) \cup \{K_{i^*,i^*+w-1}, K_{i^*+w,s}\}.$$

ii) Otherwise there are some requests mapped in  $K_{i^*,j^*}$ . Calculate  $w = j^* - i^* + 1$  and  $h = \lceil A_k / w \rceil$ . Allocate  $R = [(i^*, y'); w, h]$  for request  $A_k$ .

- **Step 4:** Update  $R_{i^*,j^*}$  or  $(R_{i^*,i^*+w-1}$  and  $R_{i^*+w,s})$ . Remove  $A_k$  from sorted  $\underline{A}$ .

In general, a burst in the DL subframe may consist of up to three contiguous parts. Part A is a rectangle region in the bottom of the burst which contains all the MUST data and possibly some WISH data or even some of the over allocated slots. Part B is the next region which contains only WISH data. Part C is a row on the top of the burst which contains over allocated slots along with the remaining WISH data, when the over allocated slots exist. Fig. 3.1 is an example of a general burst.

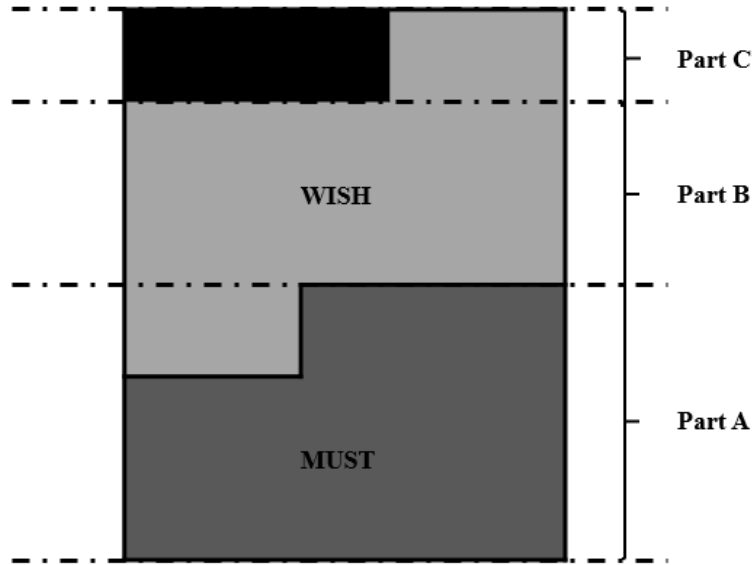


Fig. 3.1 Example of a general burst in DL sub-frame.

### 3.2.2 E-TLRM second phase

After finishing the first phase, some unmapped requests with both MUST and WISH parts might exist in sorted  $\underline{A}$ . To map as much urgent data as possible, the second

phase tries to map the MUST parts into the DL sub-frame.

Let  $\Psi = \{K_{1,p_1}, K_{p_1+1,p_2}, \dots, K_{p_l+1,p_s}\}$ ,  $1 \leq p_1 \leq p_2 \leq \dots \leq p_l \leq p_s$ , be the set of all strips after phase 1 is completed. Then Phase 2 continues through 3 steps as below:

- **Step 1:** Select sequentially unmapped requests  $A_k$  in sorted  $\mathcal{A}$  to map the MUST part  $M_k$ . Phase 2 terminates when sorted  $\mathcal{A} = \emptyset$ .

- **Step 2:** Find a strip in  $\Psi$  to map  $M_k$ .

- i) If  $M_k \leq R_{i,j}$  for some  $K_{i,j}$  in  $\Psi$ , select the strip which has the smallest available rectangle that can accommodate  $M_k$ , say  $K_{i^*,j^*}$ , to map  $M_k$ . Go to Step 3.

- ii) Otherwise, return some mapped WISH part's data so that  $M_k$  can be mapped in a strip. Let  $C_{i,j}$  and  $B_{i,j}$  be the total number of slots belonging to Part C and Part B in strip  $K_{i,j}$ , respectively.

- a. If  $M_k \leq (R_{i,j} + C_{i,j})$  for some  $K_{i,j}$  in  $\Psi$ , then select the strip with the largest  $R_{i,j}$ , say  $K_{i^*,j^*}$ . Remove row by row the slots belonging to Part C in  $K_{i^*,j^*}$  until  $M_k$  can be mapped in the strip. When removing a row in Part C, the row that has more over allocated slots will be removed first. Shift up all empty rows to the top and go to Step 3.

- b. Otherwise, if  $M_k \leq (R_{i,j} + C_{i,j} + B_{i,j})$  for some  $K_{i,j}$  in  $\Psi$ , then select the strip with largest  $R_{i,j} + C_{i,j}$ , say  $K_{i^*,j^*}$ . Remove all rows belonging to Part C in  $K_{i^*,j^*}$ . Remove row by row the slots belonging to Part B until

$M_k$  can be mapped in the strip. Shift up all empty rows to the top and go to Step 3. If  $M_k > (R_{i,j} + C_{i,j} + B_{i,j})$  for all  $K_{i,j}$ , remove  $A_k$  from sorted\_ $\mathbf{A}$  and go to Step 1.

- **Step 3:** Map  $M_k$  in  $K_{i^*,j^*}$ . If there are unused slots in  $K_{i^*,j^*}$  map the WISH part of  $A_k$ . Remove  $A_k$  from sorted\_ $\mathbf{A}$ .

### 3.3 An E-TLRM Example

Before leaving this chapter, we use a simple example to describe the operation of our proposed algorithm. Assume that there are 12 time slots and 30 sub-channels in the DL sub-frame. The Mobile WiMAX system uses 6 MCSs: QPSK 1/2; QPSK 3/4; 16QAM 1/2; 16QAM 3/4; 64QAM 2/3; 64QAM 3/4 which have number of bytes per slot are 6, 9, 12, 18, 24, 27 and are converted to 6 modulation levels from 1 to 6, respectively. There are 10 MS and in each data request, a MCS is assigned randomly, urgent data size equal to 240 Bytes, and non-urgent data size is assigned randomly from 0 to 1520 Bytes. The set of all data request calculated in number of slots are shown in **Table 3.1**.

**Table 3.1 An example of proposed algorithm**

Resources	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
(Slots)	32	39	71	49	150	92	27	14	162	98
$m_i$	5	3	2	4	1	3	6	4	1	2
MUST part	10	20	27	14	40	20	9	14	40	27
WISH part	22	19	44	35	110	72	18	0	122	71



In the first phase, after sorting the set of requests we have sorted  $\mathcal{A} = \{A_7, A_1, A_4, A_8, A_6, A_2, A_{10}, A_3, A_9, A_5\}$ . The first iteration begins with selecting the first strip  $K_{1,12}$  that have the largest available space  $R_{1,12} = [(1,1);12,30]$  to map the highest-priority request  $A_7 = 27$  ( $m_7 = 6$ ). According to Step 3, we calculate  $w = \lceil 27/30 \rceil = 1$ ,  $h = \lceil 27/1 \rceil = 27$ , then the rectangle  $R = [(1,1);1,27]$  is allocated to  $A_7$ .  $K_{1,12}$  will be split into  $K_{1,1}$  and  $K_{2,12}$ . The set  $\Psi$  will be updated as  $\Psi = \{K_{1,1}, K_{2,12}\}$ . In the second iteration, the maximum rectangle  $R_{2,12} = [(2,1);12,30]$  in  $K_{2,12}$  is selected to map  $A_1 = 32$ . According to Step 3, the rectangle  $R = [(2,1);2,16]$  is allocated to  $A_1$ .  $K_{2,12}$  is split into  $K_{2,3}$  and  $K_{4,12}$ , and the set  $\Psi$  is updated as  $\Psi = \{K_{1,1}, K_{2,3}, K_{4,12}\}$ . The first phase is executed for 4 more iterations to map requests  $A_4, A_8, A_6$ , and  $A_2$ . After finishing the first phase, we have sorted  $\mathcal{A} = \{A_{10}, A_3, A_9, A_5\}$  and  $\Psi = \{K_{1,1}, K_{2,3}, K_{4,5}, K_{6,9}, K_{10,11}, K_{12,12}\}$ .

In the first iteration of the second phase, we try to map  $M_{10}$  into  $K_{i,j}$ , for all  $K_{i,j} \in \Psi$ . According to Step 2i) and Step 3, the rectangle  $R_{6,9} = [(6,24);4,7]$  in  $K_{6,9}$  is used to allocate  $M_{10}$ . Similarly, in the second iteration, the rectangle  $R_{12,12}$  in  $K_{12,12}$  is used to allocate  $M_3$  (and a part of  $W_3$ ). In the third and fourth iteration, we choose  $K_{10,11}$  and  $K_{4,5}$  to map  $M_9$  and  $M_5$  by returning the over allocated slots and some of the WISH parts of  $A_2$  and  $A_4$ , respectively. The result is shown in Fig. 3.2.

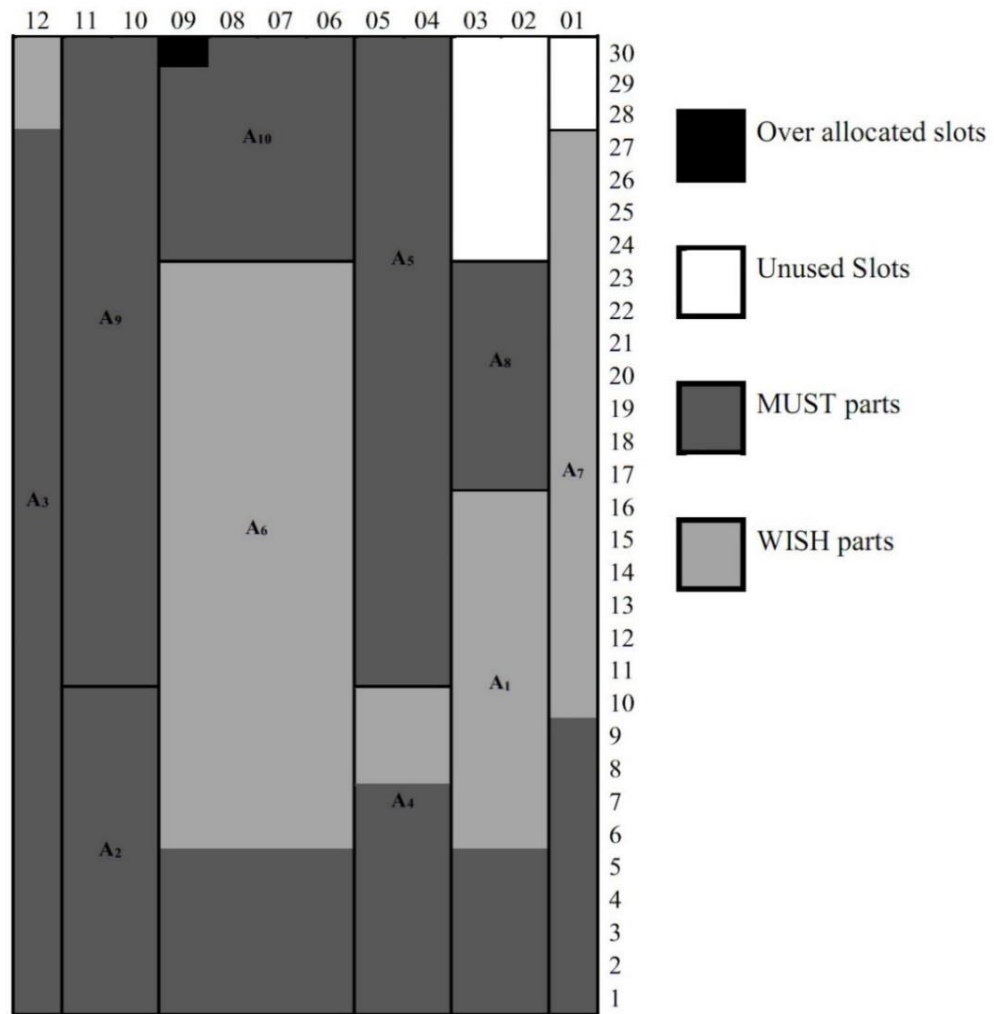


Fig. 3.2 An Example of DL data mapping using E-TLRM algorithm.

For this example, the MUST part, WISH part and unused slots are shown in dark gray, gray and white, respectively. The total number of over allocated slots (shown in black) is 1. All of the MUST data are mapped. The efficiency (percentage of used space over total space) of the proposed algorithm is 95%, with over allocated slots and unused slots being counted as wasted space. The final results of this example are shown as **Table 3.2**.

**Table 3.2 E-TLRM example results**

<b>Over allocated Slots</b>	<b>1</b>
<b>Unused Slots</b>	<b>17</b>
<b>Unmapped MUST parts</b>	<b>0</b>
<b>Efficiency</b>	<b>95%</b>



# Chapter 4 Performance Evaluation

In this chapter, we present the simulation results to show numerical results comparing our proposed algorithm E-TLRM with eOCSA and TLRM algorithms. In our simulations, we assume that the request for each MS is randomly generated; with the constraint that the urgent and non-urgent traffics (in Bytes) are generated uniformly in  $[0,240]$  and in  $[0,1520]$  with different distributions and are converted to MUST parts and WISH parts (in slots), respectively. We also assume that the MCS of each MS is assigned randomly and one burst in DL sub-frame is allocated for each MS.

The number of MSs is randomly chosen from 1 to 36. The over allocated slots, unused slots, efficiency, number of mapped request's MUST part and urgent data throughput are averaged out over 10000 trials. For our simulation, the physical bandwidth is 10 MHz and the frame duration is 5 ms. The duplexing technique is TDD, and the permutation mode is PUSC, which is the most commonly used mode. The simulations use 6 MCSs: QPSK 1/2; QPSK 3/4; 16QAM 1/2; 16QAM 3/4; 64QAM 2/3; 64QAM 3/4 which have number of bytes per slot are 6, 9, 12, 18, 24, 27 and are converted to 6 modulation levels from 1 to 6, respectively. A summarized description of the system simulation parameters are shown in **Table 4.1**.

**Table 4.1 System simulation parameters**

Parameter	Value
Frame length	5 ms
Channel BW	10 MHz
Permutation scheme	PUSC
Number of sub-channels	30
DL sub-frame	12 slot columns
Total number of slots per DL sub-frame	30 x 12 slots
Modulation and Coding scheme (bytes/slot)	QPSK 1/2( $\mu_1=6$ ) QPSK 3/4( $\mu_2=9$ ) 16QAM 1/2( $\mu_3=12$ ) 16QAM 3/4( $\mu_4=18$ ) 64QAM 2/3( $\mu_5=24$ ) 64QAM 3/4( $\mu_6=27$ )
Urgent data	0 ~240 Bytes
Non-urgent data	0 ~1520 Bytes
Simulation time	10000 frames

As mention before, the selected algorithms to compare our E-TLRM algorithm with is that of eOCSA and TLRM. We start by evaluating the impact of unused slots, which is defined as the fraction of the DL sub-frame taken by the slots that are left unused in certain frame.

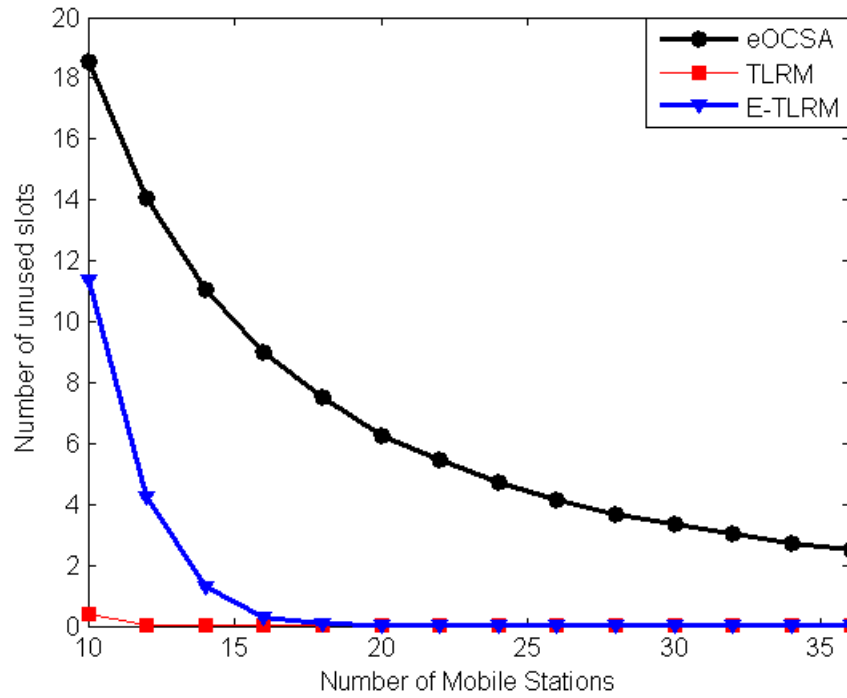


Fig. 4.1 Average unused slots vs. number of MSs.

Fig. 4.1 shows that under the low traffic load (10 MSs), the E-TLRM algorithm generates more unused slots than TLRM algorithm. In the proposed algorithm, the requests with higher MCS levels that could be require smaller number of slots are mapped earlier. Consequently, unmapped requests in the second phase usually are of larger sizes and require some allocated WISH parts to be returned in order to map them into DL sub-frame. On average, the unused slots of E-TLRM are 11.385 and 0.392 with TLRM algorithm. When the number of MSs is high, the average of unused slots for both proposed algorithm and TLRM algorithm approach 0. In both cases, the average number of unused slots for the eOCSA algorithm is always higher than that of E-TLRM and TLRM.

In Fig. 4.2, we compare the over allocated slots of proposed algorithm with eOCSA and TLRM algorithms. As mentioned before, the over allocated slots is defined as the fraction of the DL sub-frame taken by the portions of the burst that are not actually

being utilized for data request.

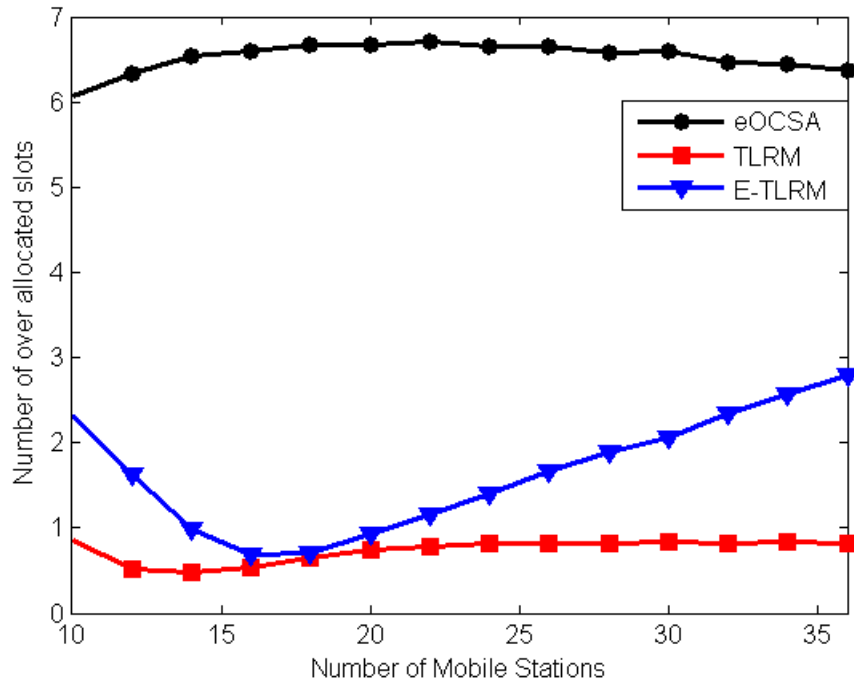


Fig. 4.2 Average over allocated slots vs. number of MSs.

Since the proposed algorithm maps the requests with higher MCS level in the first phase, the number of strips is more than that of TLRM algorithm. As a result, our proposed E-TLRM algorithm generates more over allocated slots than TLRM algorithm as shown in Fig. 4.2.

From the analyses with average of unused slots and over allocated slots, it is easy to understand why the efficiency of the proposed algorithm is slightly worse than TLRM algorithm but is still better than eOCSA as can be seen in Fig. 4.3. The average efficiency is defined as allocated slots divided by total slots per frame.

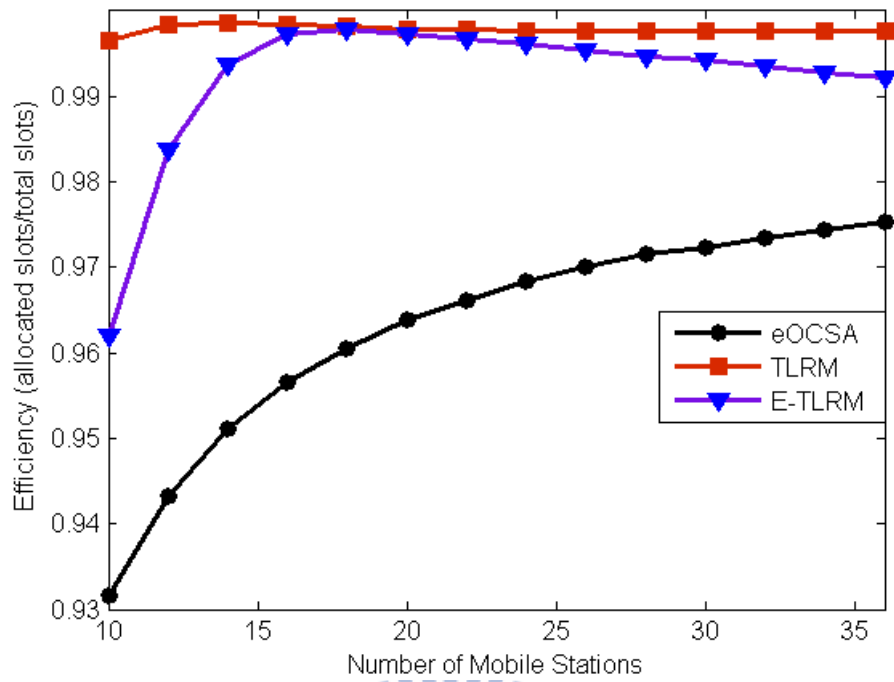


Fig. 4.3 Average efficiency vs. number of MSs.

Furthermore, we compare the throughput and average amount of mapped MUST parts of proposed algorithm with eOCSA and TLRM algorithms.

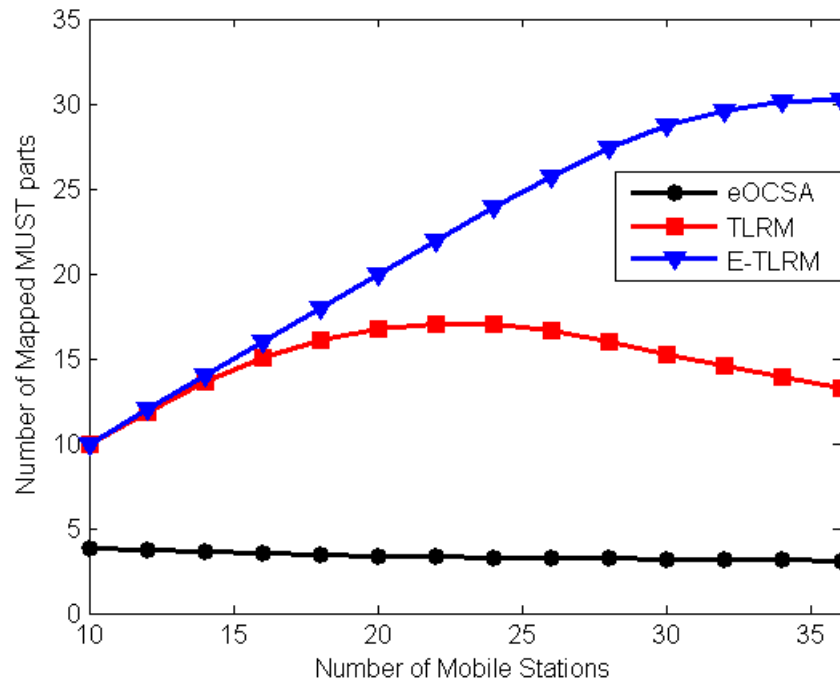


Fig. 4.4 Average number of mapped MUST parts vs. number of MSs.



Since eOCSA only considers the numbers of slots, it is possible that a MS with a worst MCS gets a highest priority and consumes a lot of slots. Without considering the traffic differentiation, the MUST part throughput of eOCSA saturates when the number of MSs is about 5, resulting in a flat curve in. TLRM can return some slots of WISH parts to allocate more MUST parts. We can see that the MUST part and throughput increases dramatically. Because E-TLRM determines the mapping order according to the MUST part's size and MCS of each MS, it outperforms both eOCSA and TLRM algorithms in terms of the throughput and the average amount of mapped MUST parts.

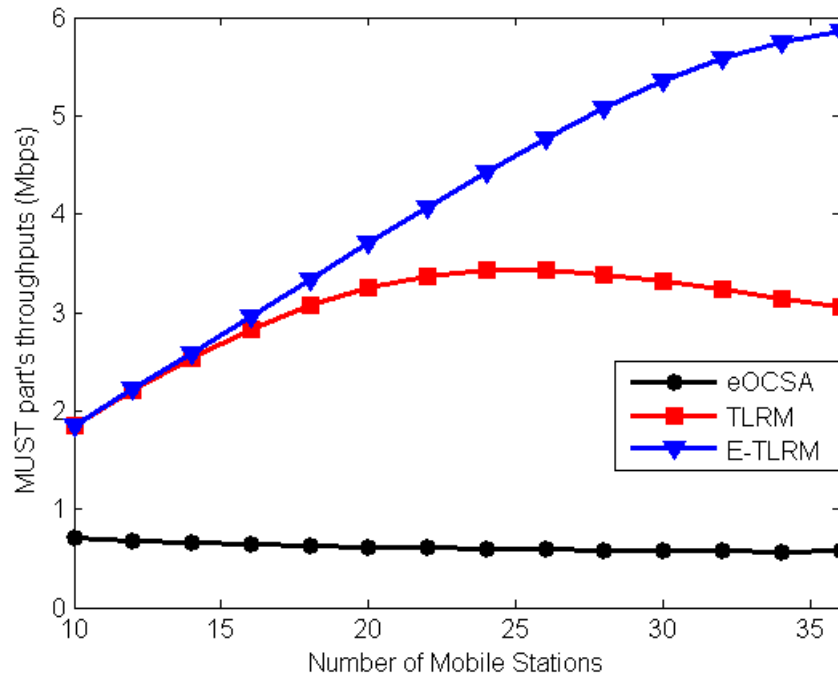


Fig. 4.5 Average mapped MUST part's throughput vs. number of MSs.

# Chapter 5 Conclusions

The main contribution of this thesis is to implement an enhanced version of the data mapping algorithm for two-level requests in IEEE 802.16e mobile WiMAX systems. We first introduce the data mapping problem in WiMAX system in Chapter 1. Then we briefly introduce the eOCSA and TLRM algorithms which proposed in literature to solve the data mapping problem in Chapter 2.

In Chapter 3, our proposed E-TLRM algorithm is described. E-TLRM, similar to previous algorithm TLRM, consists of two phases and can effectively prioritize the urgent traffic. The basic change of our proposed algorithm is that the mapping order is determined based on both the MCS and MUST part's sizes. Simulation results in Chapter 4 show that the proposed algorithm outperforms previous schemes. The proposed algorithm can serve a larger number of MUST data requests and in the meantime achieves higher throughput than previous schemes.

## Bibliography

- [1] IEEE 802.16e-2005, “IEEE Standard for Local and metropolitan area networks - Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems - Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1”, February 2006.
- [2] Cristina Ciochina and Hikmet Sari, “A Review of OFDMA and Single-Carrier FDMA and Some Recent Results”, in *Advances in Electronic and Telecommunications*, vol. 1, no. 1, April 2010.
- [3] Krishna Balachandran, Doru Calin, Fang-Chen Cheng, Niranjana Joshi, Joseph H. Kang, Achilles Kogiantis, Kurt Rausch, Ashok Rudrapatna, James P. Seymour, and Jonqyin Sun, “Design and Analysis of an IEEE 802.16e-Based OFDMA Communication System”, in *Bell Labs Technical Journal*, vol. 11, no. 4, pp. 53-73, 2007.
- [4] WiMAX Forum, “WiMAX System Evaluation Methodology V2.1,” Jul. 2008, 230 pp. URL=<http://www.wimaxforum.org/technology/documents>.
- [5] Yanqun Le, Yi Wu, and Dongmei Zhang, “An Improved Scheduling Algorithm for rtPS Services in IEEE 802.16”, *IEEE VTC Spring 2009*.
- [6] 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.
- [7] 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.
- [8] I. Gutiérrez, F. Bader, R. Aquilué, J. Pijoan, “Contiguous Frequency-Time

Resource Allocation and Scheduling for Wireless OFDMA System with QoS Support”, EURASIP Journal on Wireless Communications and Networking Volume 2009, Article ID 134579, pages 12.

- [9] T. Ali-Yahiya, A. Beylot, and G. Pujole, “Downlink Resource Allocation Strategies for OFDMA based Mobile WiMAX”, Telecommunication System, vol 44, no. 1-2, pp. 29-37, June 2010.
- [10] A. Bacioccola, C. Cicconetti, L. Lenzini, E. Mingozzi, and A. Ertar, “A Downlink Data Region Allocation Algorithm for IEEE 802.16e OFDMA”, 6th International Conference on Information, Communication and Signal Processing, pp. 1-5, December 2007.
- [11] T. Ohseki, M. Morita, and T. Inoue, “Burst Construction and Packet Mapping Scheme for OFDMA Downlinks in IEEE 802.16 System”, Proceeding of IEEE Global Communications Conference (IEEE GLOBECOM), 2009.
- [12] Tsern-Huei Lee, Chi-Hsien Liu, Arleth Soleiy Garth Campbell, and Yaw-Wen Kuo, “A Data Mapping Algorithm for Two-Level Requests in WiMAX Systems”, IEEE VTC Spring 2012.
- [13] C. So-In, R. Jain, and A. Al Tamimi, “OCSA: An Algorithm for Burst Mapping in IEEE 802.16e Mobile WiMAX Networks”, Proceedings of the 15th Asia-Pacific Conference on Communications (APCC 2009), Oct. 2009.
- [14] C. So-In, R. Jain, and A. Al Tamimi, “eOCSA: An Algorithm for Burst Mapping with Strict QoS Requirements in IEEE 802.16e Mobile WiMAX Networks”, Proceedings of the 2nd IFIP Conference on Wireless Days, pp. 204-208, 2009.