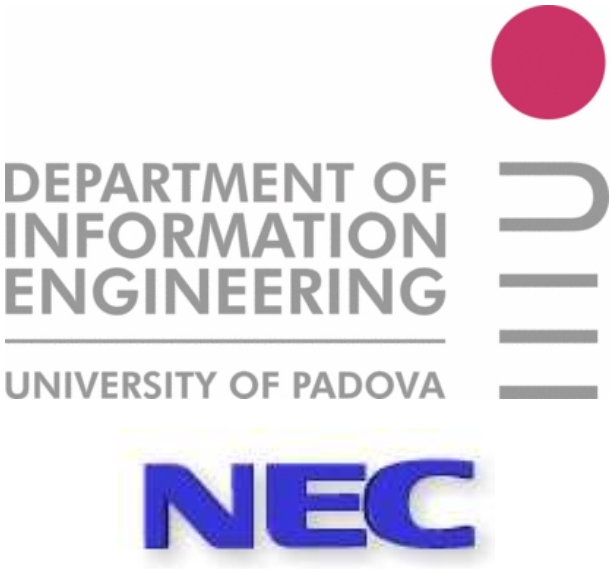


Department of Information Engineering
University of Padova

Master Project

Capacity Modeling for Admission Control in WiMAX Networks

Author:
Marco Mezzavilla

University of Padova Advisor:
Prof. Michele Zorzi

NEC Supervisor:
Dr. Xavier Pérez-Costa

March 9, 2010

Abstract

WiMAX networks support QoS reservation of resources by allowing a new flow to apply for admittance in the system. Thus, there is a need for an accurate estimation of the available capacity to be shared by incoming connections. Admission control algorithm must ensure that, when a new QoS resource reservation is accepted, reservations already present in the system continue having their QoS guarantees honored. Its efficiency is then expressed in terms of accuracy and computational complexity which is the focus of the work in this thesis.

Different approaches are presented to compute the aggregated allocated capacity in WiMAX networks and, based on their limitations, the E-Diophantine solution has been proposed. The mathematical foundations for the designed approach are provided along with the performance improvements to be expected, both in accuracy and computational terms, as compared to three alternatives of increasing complexity. The different solutions considered are validated and evaluated with OPNET's WiMAX simulator in a realistic scenario. Finally, the multi-hop relay case is analyzed: a capacity model description is provided together with a conjectured reuse of the admission control algorithm designed.

Preface

This Thesis is submitted as partial fulfilment of the requirements of the Master's degree in Telecommunication Engineering at the Department of Information Engineering of the University of Padova. The duration of this thesis was nine months, starting from December 1st 2009 to August 31st 2009 in the Network Division of the NEC E Network Laboratories in Heidelberg (DE). The research carried out led to the publication of the paper 'E-Diophantine - An Admission Control Algorithm for WiMAX Networks' to the IEEE WCNC 2010 conference, and to the patent application 'A Novel Probalistic Data Structure Supporting Wildcard, Cardinality and Threshold Queries'. At NEC the project was led and supervised by Dr. Xavier Pérez-Costa. Prof. Michele Zorzi supervised the project from the University of Padova side.

Acknowledgement

I would like to thank Dr. Xavier Pérez-Costa and the wireless team for the great support and guidance during the entire project. I also would like to thank all the other members of the NEC staff and all the NEC students that made my staying and working period in Heidelberg an interesting and enriching experience.

Special thanks to my family and all my friends that supported me throughout this challenging academic career.

Finally, I want to thank the muse that inspired this work, my *Charmin Isobel*.

Index

List of Tables

List of Figures

Chapter 1

Introduction

Based on the IEEE 802.16-2009 standard, WiMAX (Worldwide Interoperability for Microwave Access) enables rapid worldwide deployment of innovative, cost-effective, and interoperable multivendor broadband wireless access products. The standard specifies the air interface, including the medium access control layer (MAC) and physical layer (PHY), of combined fixed and mobile point-to-multipoint (PMP) broadband wireless access (BWA) systems providing multiple services, as shown in Figure 1.1. For operational frequencies from 10 to 66 GHz is specified the WirelessMAN-SC PHY, based on single-carrier modulation, while for frequencies below 11 GHz, where propagation without a direct line of sight (NLOS) must be accommodated, two alternatives are provided: WMAN-OFDM (using orthogonal frequency-division multiplexing) and WMAN-OFDMA (using orthogonal frequency-division multiple access) [1].

A WiMAX network is composed of Subscriber Stations (SS) and Base Stations (BS). Base stations fully control the access to the air interface, both in the uplink (UL) and the downlink (DL) direction, and carry out diverse functions such as handling QoS provisioning, traffic classification, tunneling of information and encryption in the WiMAX cell. Subscriber stations adhere to the instructions mandated by base stations. In order to exchange information over the air, the 802.16 family of standards offers different possibilities being time division duplexing (TDD) in combination with OFDMA the choice currently favored by most implementations. The exchange is based on WiMAX frames which are arrangements of OFDMA symbols in time and subcarriers in frequency. A frame for a WiMAX TDD OFDMA system

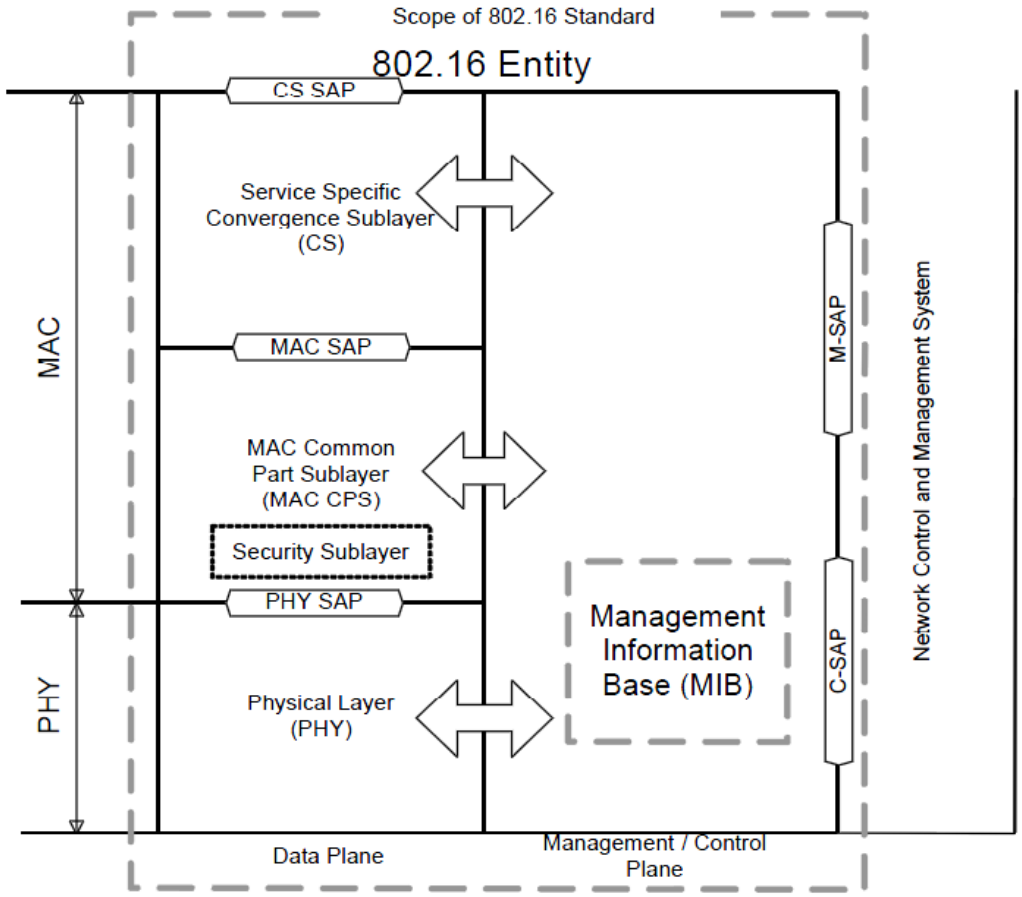


Figure 1.1: IEEE Std 802.16 Protocol reference model

is illustrated in Figure 1.2.

In TDD, a frame is divided into downlink and uplink subframe. The downlink subframe consists of a preamble followed by a frame control header (FCH), the downlink map (DL-MAP), and the uplink map (UL-MAP). The maps are used to indicate to the SSs where to find information addressed to them in the downlink or when to transmit information in the uplink. In the downlink, base stations place incoming MAC protocol data units (PDUs) in rectangular areas called bursts. The name burst comes from the fact that each rectangle can be transmitted using a particular modulation and coding scheme (MCS) referred to as burst profile. The rectangular shape restriction does not exist for uplink bursts and this considerably

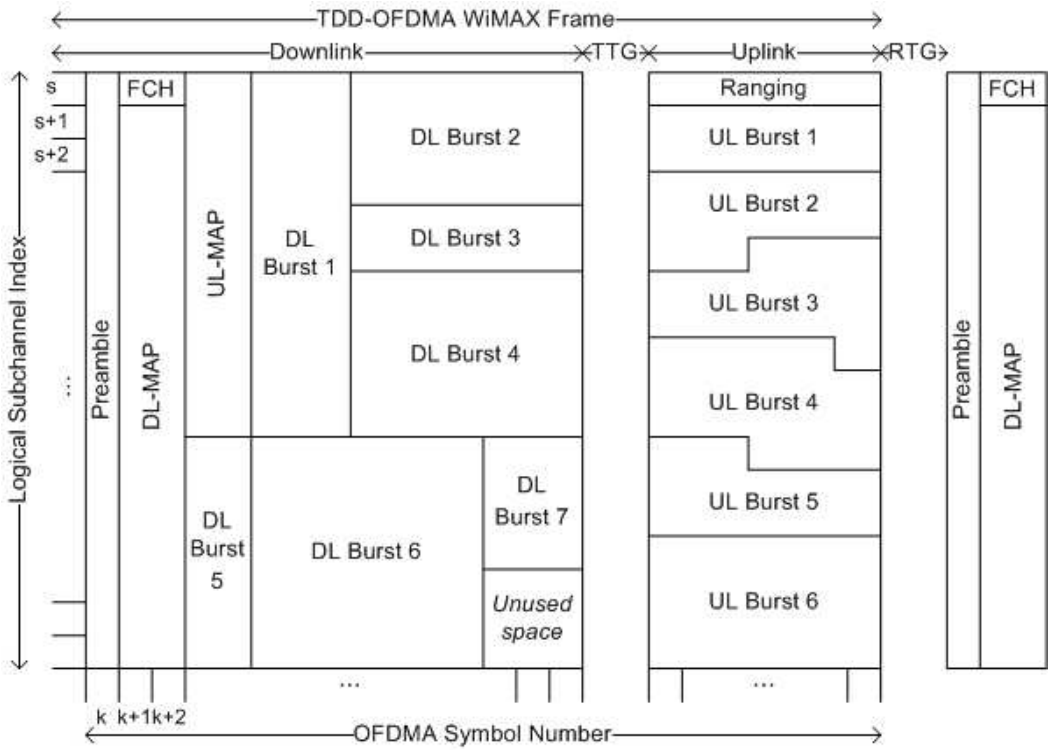


Figure 1.2: WiMAX TDD-OFDMA frame

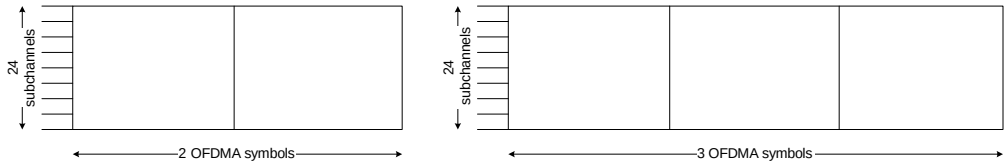
simplifies their placement in the uplink subframe. Frames can be constructed by permutating subcarriers in a variety of ways and then grouping them to create subchannels. For example, bursts inside a frame can be transmitted using non-adjacent subcarrier frequencies. This provides the system with means to counter frequency selective fading and under this scheme the subscriber stations experience a similar quality over any logical subchannel. The subcarrier groups are formed based on a set of predefined schemes. In WiMAX, such non-adjacent groupings are the partial usage of subcarrier (PUSC) mode, and the full usage of subcarrier (FUSC) mode, that are described in detail below.

The smallest atomic tile that can be assigned in a WiMAX frame for overhead or data is a slot. The definition of slot varies according to the subcarrier grouping scheme that provides, in the downlink, such a structure:

- PUSC (Partial Usage of Subcarrier): a virtual rectangle with 24 subcarriers (1 subchannel) x 2 OFDMA symbols

- FUSC (Full Usage of Subcarrier): a virtual rectangle with 48 subcarriers (2 subchannels) x 1 OFDMA symbol

The current research focuses on the PUSC case, a mandatory mode in WiMAX. In the uplink a slot corresponds to one subchannel by three OFDMA symbols, s shown in Figure 1.3.



(a) Downlink slot (b) Uplink slot

Figure 1.3: PUSC slot structure

The selection of data to be placed in a downlink subframe is performed by a QoS scheduler. Such a scheduler should be aware of the QoS requirements of the flows in the system and schedule transmissions accordingly. The data selected is then placed into the downlink subframe according to the downlink MAP packing algorithm instructions. For each burst, the DL-MAP packing algorithm is in charge of computing its appropriate dimensions and location. The area of each burst is computed taking into consideration burst profile information provided by a channel monitor. The overall performance of the system depends on the combined efficiency of the QoS scheduler and DL-MAP packing algorithm which try to maximize the radio resources usage while guaranteeing the required QoS to applications. At the MAC layer, WiMAX provides a connection oriented service in which logical connections between mobile stations and base stations are distinguished by 16 bit connection identifiers (CID). A base station assigns CIDs to unidirectional connections; this means that the identifiers for uplink and downlink are different. The MAC layer is also in charge of mapping data to the correct destination based on the CID. A mobile station will typically be assigned multiple CIDs, a primary one for management purposes and one or more secondary ones used to carry data connections. To initiate a data transfer either a mobile station or a base station creates a service flow. Independently of who requests the creation, the base station

is in charge of assigning the flow a 32 bit service flow identifier (SFID). Each admitted service flow is transported over the air using a particular CID. Additionally, any service flow is associated with a set of QoS parameters such as delay, jitter, or throughput. Service flows with the same QoS parameters are grouped into a service flow class. Classes are not defined in the standard; their definition is left to the service providers. Finally, traffic classification at the MAC layer is done based on the provider defined classes. Assuring that QoS requirements are met for all service flows that have been admitted in the system is also done at the MAC layer. Each flow in the system can negotiate, according to a set of QoS parameters, a particular scheduling service. Five different types of scheduling services are defined for WiMAX, each of them providing different type of QoS guarantees. The unsolicited grant service (UGS) is suited for flows with fixed-sized packets arriving at constant rates; it resembles wired line provisioning services such as E1 or T1. Additionally, three polling services (PS) are made available by the standard. The real time PS (rtPS) is tailored to support real time applications with packets varying in size since it is guaranteed a periodic grant to send information over the channel. The non-real time PS (nrtPS) is similar, but the polling mechanism in this case does not necessarily guarantee mobile stations a timely access to the uplink channel. A more flexible option which incorporates features from UGS and PS is the extended real-time PS (ertPS) tailored for applications with time varying bandwidth requirements. The last service is a best effort one (BE), for which no guarantees are in place. WiMAX's connectivity architecture is specified by the Network Working Group (NWG) of the WiMAX Forum in a network reference model (NRM). The architecture has three major components, the mobile stations, an access service network (ASN) and a connectivity service network (CSN). Low mobility stations can also be referred to as subscriber stations. The model components and an example of their interconnection are illustrated in Figure 1.4. In the model, an ASN contains WiMAX's base stations and gateways. Base stations control communications with mobile or fixed users, while gateways coordinate mobile access to the radio network. The CSN provides IP functionalities to subscribers and connectivity to IP networks. Each of the components of the NRM constitutes a logical entity which in practice may be implemented in one or more physical network components.

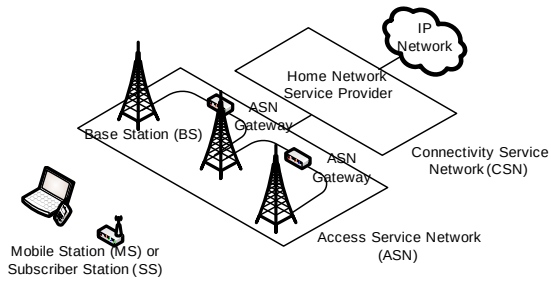


Figure 1.4: WiMAX network reference model

1.1 Capacity Modeling of WiMAX Networks

In order to be able to represent the available capacity of a WiMAX frame it will be modeled the two-dimensional capacity as a one-dimensional TDMA frame where each slot corresponds to one OFDMA slot. Multiple slots might be allocated to a given user in one frame. As it will be addressed later, capacity in bits of each slot can vary. The geometrical configuration of a slot varies depending on the frequency arrangement being adopted, as seen before.

In the following it is described in detail how to apply the two-dimensional to one-dimensional modeling of a WiMAX frame capacity differentiating between the downlink and uplink cases.

1.1.1 Downlink

The capacity of a WiMAX downlink subframe is composed in this model of three parts:

- Overhead (OH): includes Preamble, FCH and MAPs
- Data payload (D)
- Reserved (R): the reserved part can be used to accommodate changes in MCS of services already admitted in the system, retransmission needs and/or transmissions of best effort traffic.

$N_S = N_{OH} + N_D + N_R$ corresponds to the total number of available slots in the WiMAX system for data transmission between one Base Station (BS) and one

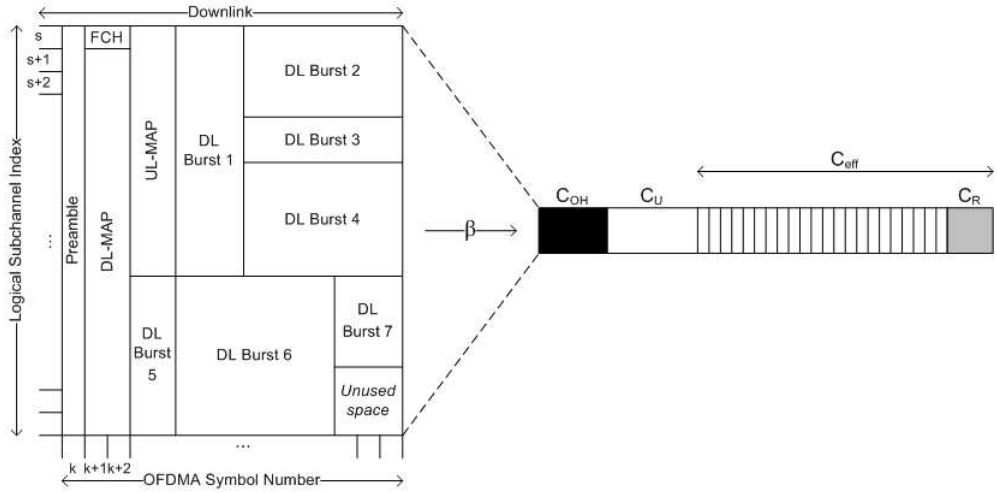


Figure 1.5: 2D to 1D downlink capacity mapping

or more subscriber stations (SSs). The number of slots available varies with the system configuration chosen.

The capacity of the system in the downlink will be here expressed as: $C = N_S/t_F$, where t_F is the frame duration (in sec). Similar definitions will be used for C_R and C_D . Values that are included in the standard for t_F are 10 ms and 5 ms.

C_{OH} corresponds to the capacity used by the overhead portion of the frame. $C_{OH} = PREAMBLE + FCH + \alpha \cdot MAP_{DL} + \gamma \cdot MAP_{UL}$. The value of α and γ varies with the number of connections being placed in the frame as well as whether traffic fragmentation or aggregation is being used.

Effective capacity is defined as $C_{eff} = \beta(C - C_{OH} - C_U)$, where β is associated to the capacity loss due to packing inefficiency and has a value range between 0 and 1 and C_U corresponds to the amount of capacity already allocated. Note that the packing loss will also vary depending on the packing arrangement (or permutation) used. For PUSC a simple packing arrangement might result for instance in $\beta = 0.6$ while more elaborate packing approaches might result in values closer to 0.8. On the other hand, when HARQ is used the packing inefficiency is zero, i.e., $\beta = 1$. See Figure 1.5 for a graphical representation of the previous definitions.

It will be assumed that each slot allows for one among a set of M transmission rates, expressed in bits per symbol, $R = \{R_1, ..., R_M\}$ and where $R_1 < R_2 < ... < R_M$ which depend on the MCS being used for a transmission. Hence, the effective

capacity at a specific slot can also be expressed in bits per second which can then be computed as $R_{eff} = C_{eff} \cdot C_{SU} \cdot R_i$

1.1.2 Uplink

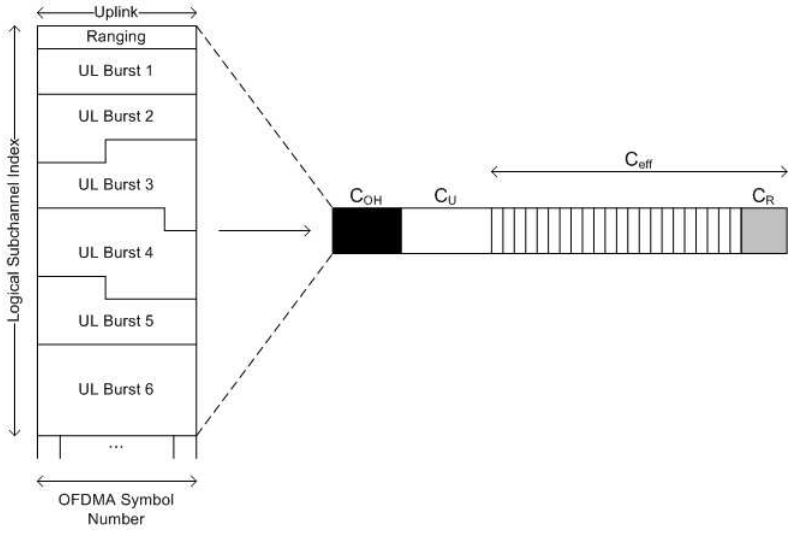


Figure 1.6: 2D to 1D uplink capacity mapping

The uplink case is very similar to the downlink with the following two main differences. First, the packing is performed in a raster and thus, no efficiency is lost due to packing, i.e., $\beta = 1$. Second, the overhead does not depend on the number of bursts transmitted but it corresponds to the capacity required for the ranging process. See Figure 1.6 for an illustration of this case.

1.2 Admission Control

WiMAX networks support QoS reservation of resources by allowing a new flow to apply for admittance in the system through a Dynamic Service Addition REQuest message (DSA-REQ). Such requests contain a QoS parameter set which includes different mandatory information depending on the data delivery service requested in the downlink or the uplink direction. Table 1.1 summarizes the required QoS parameter set per data delivery service according to the IEEE 802.16 standard [1]. Additionally, other QoS parameters can be specified to further define the QoS

guarantees required by a flow and allow for a higher efficiency of the resource utilization in the network, e.g., Maximum Sustained Traffic Rate, Traffic Priority, and so forth. A similar set of parameters is required in the uplink direction, as shown in Table 1.2.

	UGS	ERT-VR	RT-VR	NRT-VR	BE
Min. Resv. Tr. Rate (MRTR)	x	x	x	x	
Max. Sust. Tr. Rate (MSTR)		x	x	x	x
SDU size	x				
Maximum Latency	x	x	x		
Tolerated Jitter	x	x			
Traffic Priority		x		x	x
Req./Trans. Policy	x	x	x	x	x

Table 1.1: Downlink required QoS parameters per data delivery service

	UGS	ertPS	rtPS	nrtPS	BE
Min. Reserved Traffic Rate	x	x	x	x	
Max. Sustained Traffic Rate		x	x	x	
SDU size	x				
Maximum Latency	x	x	x		
Tolerated Jitter	x				
Traffic Priority				x	
Uplink Grant Sched. Type	x	x	x	x	
Request/Transmission Policy	x	x	x	x	x
Unsol. Grant/Polling Interval	x	x	x		

Table 1.2: Uplink mandatory QoS parameters per scheduling service

Admission control is always one of the most significant issues in wireless communications. The basic admission mechanisms of the guard channel and queuing were introduced in the mid-80s to give priority to handover calls over new calls [2], [3]. An admission control mechanism decides which flows may be allowed into a network without the network being saturated. It uses knowledge of incoming flows and the current network situation to ensure the QoS for flows in the network, hence the importance of its efficiency, both in terms of accuracy and computational complexity.

1.2.1 Related Work

Current literature on admission control for WiMAX proposes a wide range of options achieving very different levels of accuracy as well as computational load. The authors in [4] propose a simple approach that is mainly based on the mean data rate requirements that an application specifies. With such knowledge, connections from different services can be progressively admitted into the WiMAX system by following a predetermined priority order. Such approach requires few computational resources; however, neither does it take into consideration the time-varying nature of typical applications such as video or voice with activity detection nor the time period at which these resources are required. Thus, actual available resources might be unused. A similar solution is considered in Chapter 2 referred to as *Worst Case*. A similar approach is proposed in [5] for uplink connections. The work is extended to include a bandwidth estimation method used to monitor the queue lengths of all polling service connections at regular intervals. Such monitoring can be used to estimate dynamic bandwidth requirements. However, the granularity of the monitoring interval hinders the method from following fast changing requirements such as those found in modern video applications.

A different approach is proposed in [6] where the variance of a flow bandwidth requirements is proposed as a statistic to describe the application requirements. The authors further extend this method in [7] where they take into account the predicted fraction of packets delayed above a threshold. However, there is no proof that variance is a good descriptor for all traffic types. Such knowledge can then be used to assess if the QoS requirements for a particular flow can be fulfilled.

In [8] a fuzzy-logic based controller is employed to predict the blocking probability of a particular flow. The authors claim that the varying nature of real time applications can be taken into consideration by a 'rule-based' controller. However, a validation of such controller against diverse types of traffic is not provided. Finally, in [9] an accurate admission control algorithm for video flows is proposed which takes into account both throughput and delay requirements. However, as it will be seen in Chapter 2 for the approach referred to as *Diophantine*, it can not be used in practice due to its computational load and therefore, an alternative is needed.

1.3 Outline

This thesis examines the importance of an accurate system capacity modeling and its adjacent admission control plane. The mathematical foundations for the designed approach are provided along with the performance improvements to be expected, both in accuracy and computational terms, as compared to three alternatives of increasing complexity. The different solutions considered are validated and evaluated with OPNET's WiMAX simulator in a realistic scenario. The remainder of this thesis is organized as follows.

An overview of the WiMAX system has already been exposed in the current chapter, together with a technical description of the system capacity modeling that represents the basis of the admission control algorithm proposed, and the State of the Art relatively to WiMAX networks.

Chapter 2 analyzes the theoretical foundations of each admission control algorithm along with the mathematical modelization of the proposed solution, *E-Diophantine*, and an overall performance comparison whose details evidence the advantages coming from this work. Besides, non-ideal conditions are evaluated, such as jitter and MCS (Modulation and Coding Scheme) changes, whose impacts has been studied and commented in relation to the different admission control approaches.

Chapter 3 illustrates the efficiency of the proposed algorithm by comparing its predictive trend with the statistics collected from multiple scenarios created ad-hoc. The output graphs show the reliability of the admission control mechanism created, and the versatility of the solutions that have been analyzed in the previous chapter.

Chapter 4 is devoted to the analysis of the multi-hop relay scenarios, for which an admission control has been proposed based on the extension of the *E-Diophantine* algorithm developed and validated in the previous chapters.

Finally, the concluding remarks are summed up in Chapter 5.

Additionally, this work also includes two appendices.

Appendix A illustrates the organizational framework of the research carried out to achieve the issues detailed and evaluated throughout the thesis.

Appendix B includes the list of acronyms used in this work and their respective full-length form.

Chapter 2

Admission Control Algorithm

In order to describe the admission control system it is assumed that for each reservation i a minimum set of QoS requirements can be derived for all scheduling services but Best Effort as: given a starting time t_i, a certain amount of capacity B_i (bits) should be reserved periodically for transmitting flow's i data within a time interval T_i. Relevant examples of other wireless technologies which support reservation of resources in a similar way are 3G networks for cellular technologies and 802.11e HCCA for Wireless Local Area Networks.

Considering a new reservation i requesting acceptance in the system, an admission control algorithm has to evaluate whether there is enough capacity to admit the new reservation while still honoring the QoS of reservations already accepted. Such a resource reservation request can be modeled as a periodic discrete sequence of Kronecker deltas with amplitude B_i in the following way

$$B_i \cdot \delta_{t_i+n\cdot T_i}(t) = \begin{cases} B_i & \text{if } t = t_i + n \cdot T_i \ ; \ n \in \mathbb{Z} \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

Assuming a WiMAX system with a capacity available for data with QoS requirements C_{av} and N reservations already granted, a new reservation i can be accepted in the network if the following condition is met[1]

$$max(A(t)) \leq C_{av} \tag{2.2}$$

[1]Note that C_{av} does not necessarily have to correspond to the actual available capacity but could be a different value based on a specific operator policy.

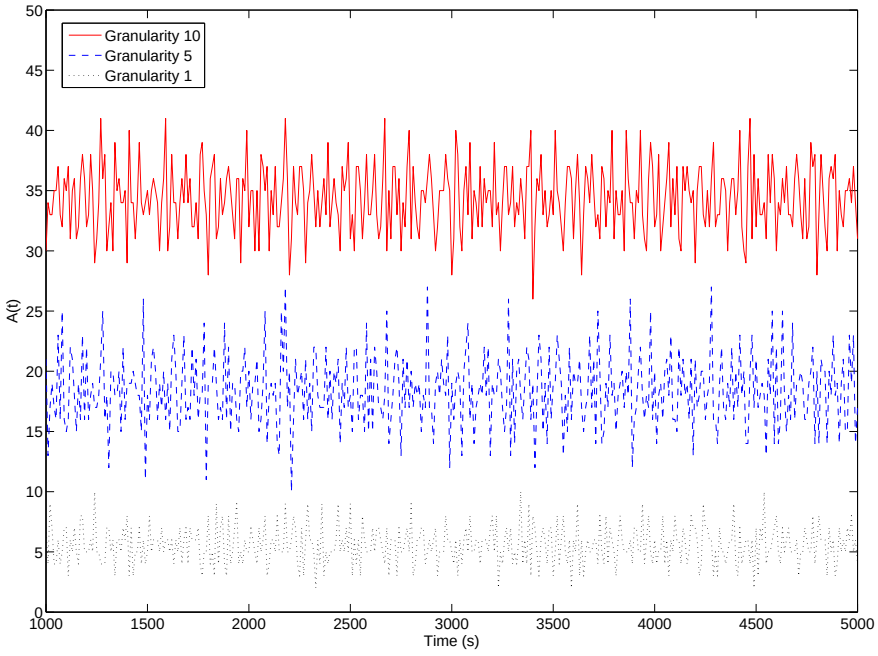


Figure 2.1: Signal envelope of 100 flows

where $A(t)$ corresponds to the aggregation, as a function of time, of the reservations of the N flows already in the system plus the one requesting admittance. See Figure 2.1 for an example of 100 reservations considering three different granularities[2] for the starting time and period of each reservation which were randomly selected from a uniform distribution between the granularity value and 100.

[2]Considering milliseconds (ms) as the units for t_i and T_i, granularity 10 would correspond to only allowing values multiples of 10 ms for t_i and T_i. In the case of granularity 5 and 1, allowed values for t_i and T_i would be multiples of 5 ms and 1 ms respectively.

2.1 Worst Case

In order to determine $max(A(t))$ different approaches can be considered. The easiest but more pessimistic approximation, hereinafter referred to as *Worst Case*, would be to assume that all admitted reservations need to be served simultaneously, i.e., without taking into account the time at which flows actually need to be served. The following equation corresponds to the *Worst Case* approximation of $A(t)$.

$$A_{worst\_case} = \sum_{i=1}^{N+1} B_i \tag{2.3}$$

Such an approach is similar to the one described in [4] and, as will be shown in Chapter 2.5 and Chapter 3, it might result in a large portion of available capacity being underutilized.

2.2 Heuristic

An accurate solution for $max(A(t))$ can be obtained by computing all values of $A(t)$ within a T_{LCM} period. Note that since $A(t)$ is composed of *N+1* periodic reservations, its period T_{LCM} corresponds to the Least Common Multiple (LCM) of the periods of the reservations in the system plus the one under consideration. This approach will be referred in the rest of the thesis as *Heuristic*. The following equation corresponds to the *Heuristic* computation of $A(t)$, and an example of a possible implementation is detailed in Algorithm 1 based on Eq. 2.4.

$$A_{heuristic}(t) = \sum_{i=1}^{N+1} B_i \cdot \delta_{t_i+n\cdot T_i}(t) \tag{2.4}$$

where LCM is the least common multiple of the periods of the N flows already accepted in the system and the new one T_{N+1}. It is to be noted that since the combination of periodic sequences with period T_i has been considered, the total combined sequence has period $LCM(T_1, T_2, ...T_{N+1})$.

The *Heuristic* approach though has a dependence with the LCM of the reservations in the system which, depending on the granularity allowed, might increase exponentially with the number of reservations and thus, become too expensive in computational terms. Therefore, such a solution is in general not feasible in practice

Algorithm 1 Heuristic algorithm to find out the maximum resource requirement within a T_{LCM} interval for a new reservation r_{N+1} with starting time t_{N+1}, period T_{N+1} and requirement B_{N+1} considering the set of N reservations already accepted in the system with their corresponding starting times $t = (t_1...t_N)$, periods $T = (T_1...T_N)$ and requirements $B = (B_1...B_N)$

Variables initialization
$T_{LCM} = 1$
Call executed for each new reservation request
$T_{LCM} = compute\_lcm(T_{LCM}, T_{N+1})$
for $i = 1$ to $N + 1$ **do**
  for $j = t_j$ to T_{LCM} **do**
    $A(j) = A(j) + B_i$
    $j = j + T_j$
  end for
end for
if $find\_maximum(A) \leq C_{av}$ **then**
  return $accept\_request(r_{N+1})$
else
  return $reject\_request(r_{N+1})$
end if

unless limitation in the granularity of periods is imposed, as complexity increases with the number of flows according to $O(LCM)$.

2.3 Diophantine

In order to remove the LCM dependency with the *Heuristic* approach, another solution is considered based on *Diophantine*[3] theory which, in general, deals with indeterminate polynomial equations that allows variables to be integers only. In the rest of the thesis this approach will be referred to as *Diophantine* and, as indicated in Chapter 1.2.1, it has already been considered as a solution for admission control in WiMAX networks [9].

The *Diophantine* solution is defined as follows. Considering a flow already accepted in the system described with the resource reservation $B_i \cdot \delta_{t_i+n_i \cdot T_i}(t)$ and a new flow requesting admittance characterized by $B_j \cdot \delta_{t_j+n_j \cdot T_j}(t)$, the maximum

[3]Diophantine equations are named after Diophantus of Alexandria, an Hellenistic mathematician of the 3rd century who studied such equations.

resource requirement, $B_i + B_j$, will occur for the set of n_i and n_j combinations which fulfill

$$\{t_i + n_i \cdot T_i = t_j + n_j \cdot T_j\} \tag{2.5}$$

where n_i and $n_j \in \mathbb{Z}$

In order to find the set of solutions for n_i and n_j, hereinafter referred to as *set of intersections*, condition 2.5 can be expressed as a linear diophantine equation with two variables in the following way

$$\{n_i \cdot T_i - n_j \cdot T_j = t_j - t_i\} \tag{2.6}$$

Then, based on the linear diophantine equations theory, it is known that there will be a set of integer solutions for n_i and n_j if

$$\frac{t_j - t_i}{d} \in \mathbb{Z} \tag{2.7}$$

where $d = gcd(T_i, T_j)$ and gcd stands for greatest commom divisor.

When the previous condition holds, the set of solutions corresponding to a specific pair of reservations can be found with the extended Euclidean algorithm which will find a and b such that

$$a \cdot T_i + b \cdot T_j = d \tag{2.8}$$

where a and $b \in \mathbb{Z}$

By applying the *Diophantine* solution to all pairs of reservations in the system, as well as to their found solutions in a recursive manner, an exact solution for $A(t)$ can be found which is independent of the LCM length.

In algorithm 2 is detailed a possible implementation of this solution where the pseudo-command $solution\_exists(.)$ corresponds to validating condition 2.7, while $find\_inters\_diophantine(.)$ corresponds to applying the extended euclidean algorithm according to Eq. 2.8 and $group\_intersections(.)$ is a function that groups new intersections found with previously found ones if they belong to the same family of solutions.

Algorithm 2 Diophantine algorithm to find out the maximum resource requirement for a new reservation r_{N+1} with starting time t_{N+1}, period T_{N+1} and requirement B_{N+1} considering the set of N reservations already accepted in the system with their corresponding starting times $t = (t_1...t_N)$, periods $T = (T_1...T_N)$ and requirements $B = (B_1...B_N)$

```
1: Call executed for each new reservation request
2: potential_intersections = true
3: inters_length = N + 1
4: while potential_intersections do
5:     for i = 1 to inters_length do
6:         for j = i + 1 to inters_length do
7:             if solution_exists(t_i, t_j, T_i, T_j) then
8:                 intersections ← find_inters_dioph(t_i, t_j, T_i, T_j)
9:             end if
10:        end for
11:    end for
12:    if is_empty(intersections) then
13:        potential_intersections = false
14:    else
15:        intersections ← group_intersections(intersections)
16:        inters_length = length(intersections)
17:    end if
18: end while
19: if find_maximum(intersections, B) ≤ C_av then
20:    return accept_request(r_{N+1})
21: else
22:    return reject_request(r_{N+1})
23: end if
24: return find_maximum(intersections, B)
```

The *Diophantine* solution though requires to compute the *gcd* for all pairs of reservations in the system as well as the sets of intersections found. As a result, its computational complexity increases significantly as the number of reservations grows and, as in the case of the *Heuristic* solution, it might become unfeasible in practice.

2.4 E-Diophantine

Based on the feasibility issues identified for both the *Heuristic* and the *Diophantine* solutions, an enhancement of the *Diophantine* approach is proposed, hereinafter referred as *E-Diophantine*, which achieves the same accuracy ast the *Diophantine* in finding the maximum of $A(t)$ but at a much lower computational cost.

The *E-Diophantine* solution proposed consists in first, exactly as in the *Diophantine* case, finding the set of intersections for all flows under consideration applying condition 2.5 and Eq. 2.8. These results are summarized in a matrix of intersections of flows as the one shown in Table 2.1 for a 10 flows example. Then, the rest of the set of intersections between the solutions found is derived based on the information obtained regarding the flows involved in each intersection set. In the following are provided the theorems and their proofs that enable the designed *E-Diophantine* algorithm.

2.4.1 Mathematical Model

Intersection of 2 Sets of Intersections

Theorem 1. For any pair of sets of intersections of 2 reservations found, they will intersect if both solutions have one reservation in common and the other two reservations intersect between each other.

Proof. Consider that for reservations i and j a set of intersections exists defined as

$$\{t_{ij} + n_{ij} \cdot T_{ij}\} \tag{2.9}$$

$$\text{where } T_{ij} = lcm(T_i, T_j) \text{ and } n_{ij} \in \mathbb{Z}$$

such that the smallest n_i and $n_j \in \mathbb{Z}$ satisfy

$$t_i + n_i^{min} \cdot T_i = t_j + n_j^{min} \cdot T_j \equiv t_{ij} \tag{2.10}$$

Then, consider another set of intersections for reservations j and k defined as

$$\{t_{jk} + n_{jk} \cdot T_{jk}\} \tag{2.11}$$

A set of intersections between both intersections sets found will exist if a set of n_{ij} and $n_{jk} \in \mathbb{Z}$ such that

$$\{t_{ij} + n_{ij} \cdot T_{ij} = t_{jk} + n_{jk} \cdot T_{jk}\} \tag{2.12}$$

Considering that t_{ij} and t_{jk} can be expressed as $t_j + n_j^{min} \cdot T_j$ and $t_j + n_j'^{min} \cdot T_j$ respectively, Eq. 2.12 can expressed as follows

$$\left\{ n_j^{min} + n_{ij} \frac{T_i}{gcd(T_i, T_j)} = n_j'^{min} + n_{jk} \frac{T_k}{gcd(T_j, T_k)} \right\} \tag{2.13}$$

Then, since

$$\left\{ n_j^{min} + n_{ij} \frac{T_i}{gcd(T_i, T_j)} \right\} \subseteq \{t_i + n_i \cdot T_i\} \tag{2.14}$$

and

$$\left\{ n_j'^{min} + n_{jk} \frac{T_k}{gcd(T_j, T_k)} \right\} \subseteq \{t_k + n_k \cdot T_k\} \tag{2.15}$$

a solution will exist for n_{ij} and $n_{jk} \in \mathbb{Z}$ such that the condition in Eq. 2.13 holds if reservations i and k intersect. The resulting set of intersections for reservations i, j and k would be then defined as

$$\{t_{ijk} + n_{ijk} \cdot T_{ijk}\} \tag{2.16}$$

$$\text{where } T_{ijk} = lcm(T_i, T_j, T_k) \text{ and } n_{ijk} \in \mathbb{Z}$$

□

Intersection of N+1 Sets of Intersections

Theorem 2. For any set of intersections of N sets of intersections found, it will intersect with another set of intersections if and only if all reservations involved in both sets of intersections intersect with each other.

Proof. Assuming a set of intersections of N sets of intersections defined as[4]

$$\{t_{1-N} + n_{1-N} \cdot T_{1-N}\} \equiv I_N \tag{2.17}$$

For a set of intersections $t_{N+1} + n_{N+1} \cdot T_{N+1} \equiv I_{N+1}$ to intersect with I_N, a set of n_{1-N} and $n_{N+1} \in \mathbb{Z}$ should exist such that

$$\{t_{1-N} + n_{1-N} \cdot T_{1-N} = t_{N+1} + n_{N+1} \cdot T_{N+1}\} \tag{2.18}$$

Considering that

$$I_N = \{t_1 + n_1 \cdot T_1\} \cap ... \cap \{t_N + n_N \cdot T_N\} \tag{2.19}$$

Then, the set of intersections I_{N+1} will intersect with I_N if and only if

$$I_{N+1} \cap \{t_1 + n_1 \cdot T_1\} \cap ... \cap \{t_N + n_N \cdot T_N\} \notin \emptyset \tag{2.20}$$

□

2.4.2 E-Diophantine Algorithm

Algorithm 3 details the steps followed by the *E-Diophantine* solution. The first part of the algorithm, which finds the first set of intersections, is identical to the *Diophantine* algorithm. Once the first set of intersections has been obtained, a matrix of intersections is computed. This operation corresponds to the function *compute_matrix_inters*(.) in Algorithm 3. Table 2.1 provides an example of a matrix of intersections found for a set of 10 flows. Such matrix of intersections can be obtained by simply traversing for each pair of flows the set of intersections obtained in the first part of the algorithm.

Based on the matrix of intersections, the *E-Diophantine* algorithm finds the rest of additional intersections by traversing for each flow the matrix of intersections and discarding the non-possible solutions by applying Theorems 1 and 2. This operation corresponds to the function *compute_inters_inters*(.). Figure 2.8 illustrates the tree of solutions found based on the matrix of solutions shown in Table 2.1.

[4]Note that the notation for a set of intersections has been simplified for readibility reasons such that a set of intersections involving several reservations is referred with a single subindex instead of with the indexes of the reservations involved.

Algorithm 3 E-Diophantine algorithm to find out the maximum resource requirement for a new flow with starting time t_{N+1}, period T_{N+1} and requirement B_{N+1} considering the set of N flows already accepted in the system with their corresponding starting times $t = (t_1...t_N)$, periods $T = (T_1...T_N)$ and requirements $B = (B_1...B_N)$

```
1: Call executed for each new flow request
2: for i = 1 to N + 1 do
3:     for j = i + 1 to N + 1 do
4:         if solution_exists(t_i, t_j, T_i, T_j) then
5:             intersections ← find_inters_dioph(t_i, t_j, T_i, T_j)
6:         end if
7:     end for
8: end for
9: intersections ← group_intersections(intersections)
10: m_inters = compute_matrix_inters(intersections)
11: for i = 1 to N + 1 do
12:     solutions_tree ← compute_inters_inters(m_inters, i)
13: end for
14: return find_maximum(solutions_tree, B)
```

In the following a graphical representation of the proposed procedure is illustrated in a step-wise fashion that details the algorithm implemented to achieve such an accurate prediction mechanism.

Once obtained the matrix of intersections by applying, as seen in the previous section, the *Diophantine* solution to the flows that belong to the system, the enhanced approach consists in a prediction based on the analysis of such matrix, regardless of each flow specifications (*starting time*, *period*, *requirement*), resulting in a consistent reduction of the computational load as will be shown in the next section.

Figure 2.2 depicts the tree-structure that directly derives from the matrix of intersections of Table 2.1, relatively to flow 1. The explanation of the steps followed within the algorithm is limited to this specific case because the procedure is the same for each flow, that corresponds to each line of such matrix.

Flows	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	1	0	1	1	1	1	1	1	0	1
2	0	1	1	1	1	1	1	1	0	1
3	1	1	1	1	0	0	0	1	1	0
4	1	1	1	1	1	1	1	1	1	1
5	1	1	0	1	1	0	1	1	1	0
6	1	1	0	1	0	1	0	1	1	0
7	1	1	0	1	1	0	1	1	1	0
8	1	1	1	1	1	1	1	1	1	1
9	0	0	1	1	1	1	1	1	1	1
10	1	1	0	1	0	0	0	1	1	1

Table 2.1: Example of matrix of intersections of flows

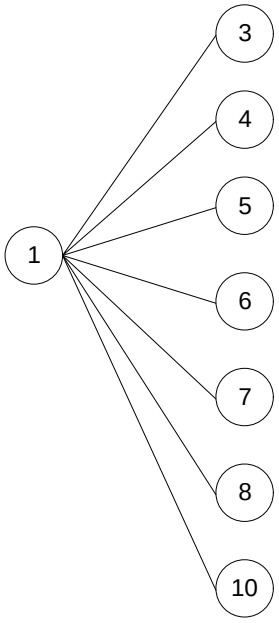
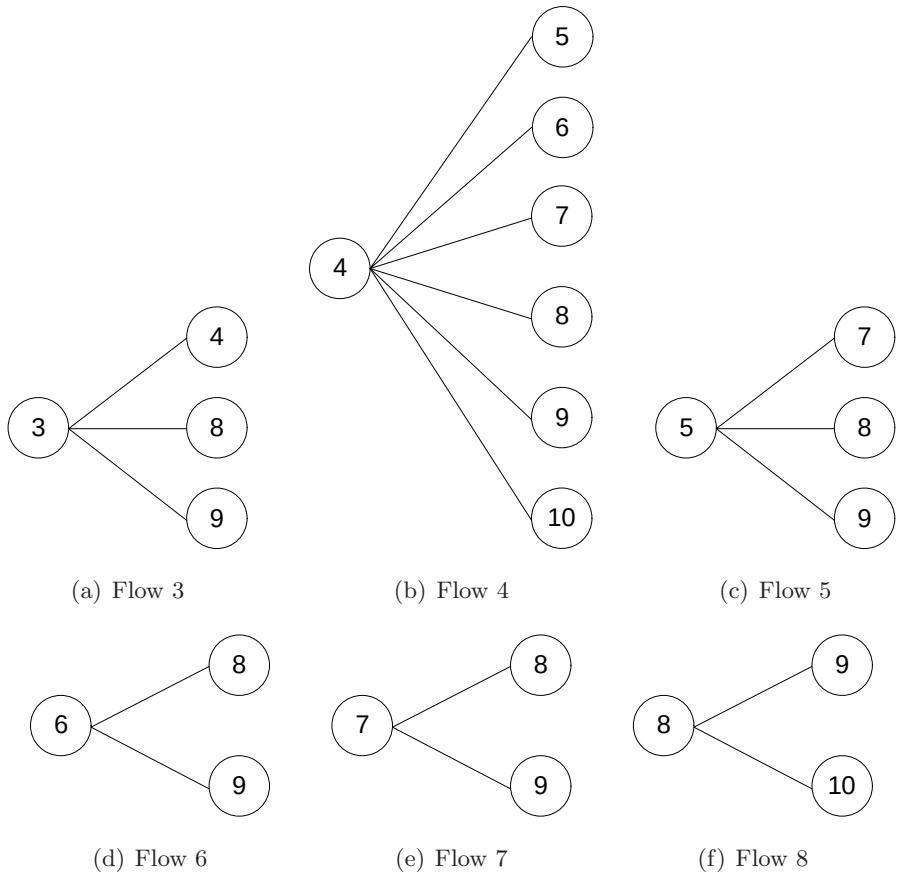


Figure 2.2: Tree of solutions - Step 1

The first step evidences the uncorrelation between flow 1 and flows 2 and 9, resulting in a lack of branches connecting them. It means that the path selected from flow 1 in order to determine the maximum resource requirement will not present any hop on flows 2 and 9 because the *Diophantine* theory evidenced the absence of any mutual intersection between them.



(a) Flow 3 (b) Flow 4 (c) Flow 5

(d) Flow 6 (e) Flow 7 (f) Flow 8

Figure 2.3: Flows mutual intersections

Figure 2.3 represents all the intersections of the flows detected in the path centered in flow 1, as depicted in Table 2.1; it will be shown in the following that, according to Theorems 1 and 2, some flows are discarded because either they do not intersect flow 1, or they do not present mutual intersections over the same branches.

Hence, as illustrated in Figure 2.4, the second step evidences the loss of intersections with flow 9, if presents, when assembling the branches of the tree centered in flow 1; this is due to the fact that, as already noticed, the ninth column (flow 9) of the first row (flow 1) of the matrix of intersections corresponds to 0, meaning that there is not intersection.

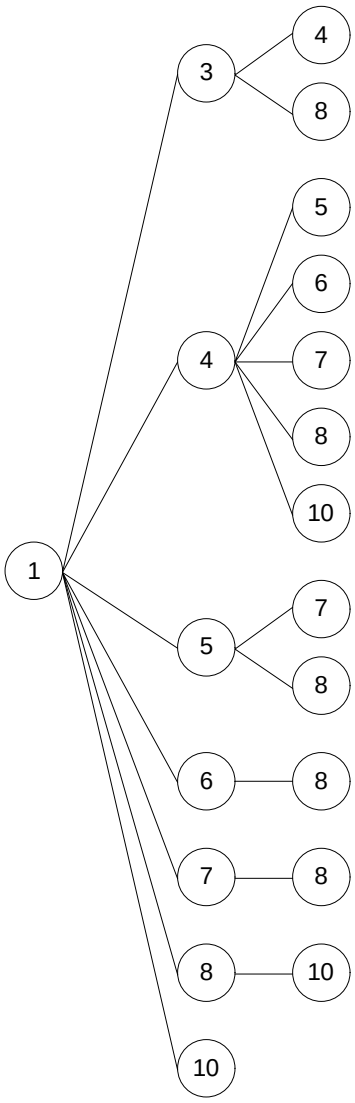


Figure 2.4: Tree of solutions - Step 2

The next step is given by the comparison between mutual intersections of the same branches. It means that a new branch is created only if the analyzed temporary position presents common intersections over the same branch. As an example, for simplicity, is taken the first branch, given by the sequence 1-3-4; as shown in Figure 2.5, a new branch has been created, returning the sequence 1-3-4-8. This is correct since the only mutual intersection between flow 4 and the set of intersections resulting from the sequence 1-3 is flow 8.

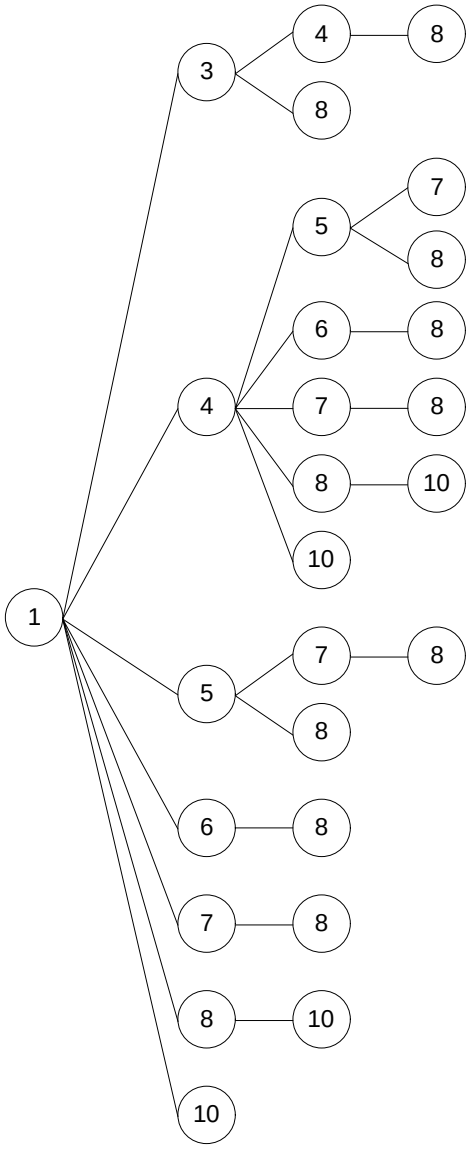


Figure 2.5: Tree of solutions - Step 3

Finally, when the mutual intersections of all the branches have been identified by iterating the procedures previously detailed, the maximum resource requirement is given by the longest sequence found. As illustrated in Figure 2.6 there will be a point in time where 5 flows intersect each other; this is the sequence 1-4-5-7-8.

It has to be noted that the procedures detailed above regard only the first line of the matrix of intersections; in order to provide the overall maximum resource requirement prediction, the algorithm has to iterates through all the lines of the matrix depicted in Table 2.1. The graphical representation of such iteration is given in Figure 2.8.

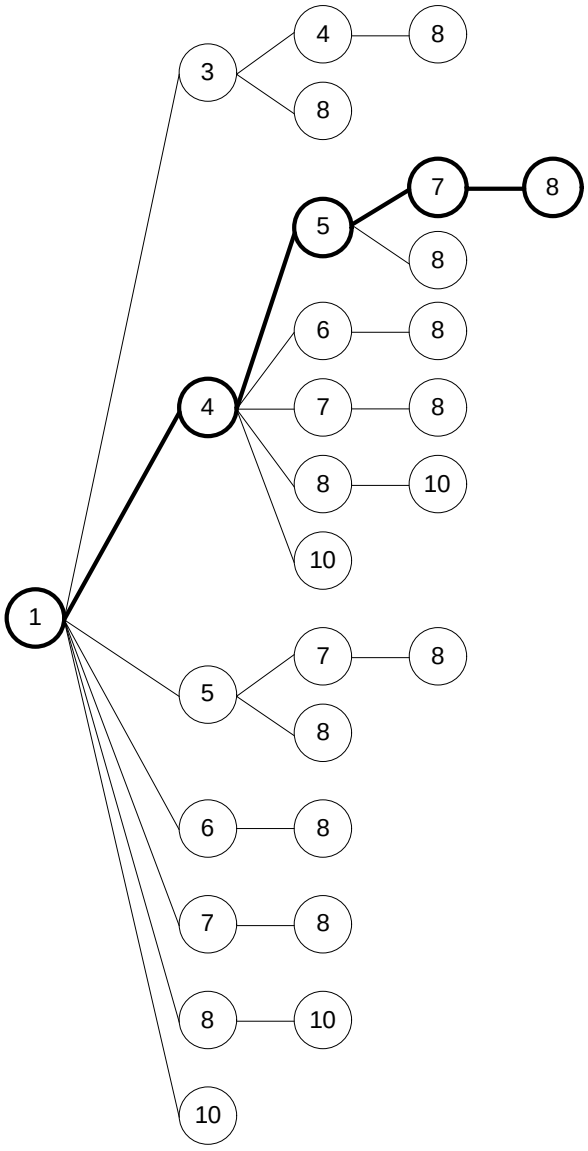


Figure 2.6: Tree of solutions - Step 4

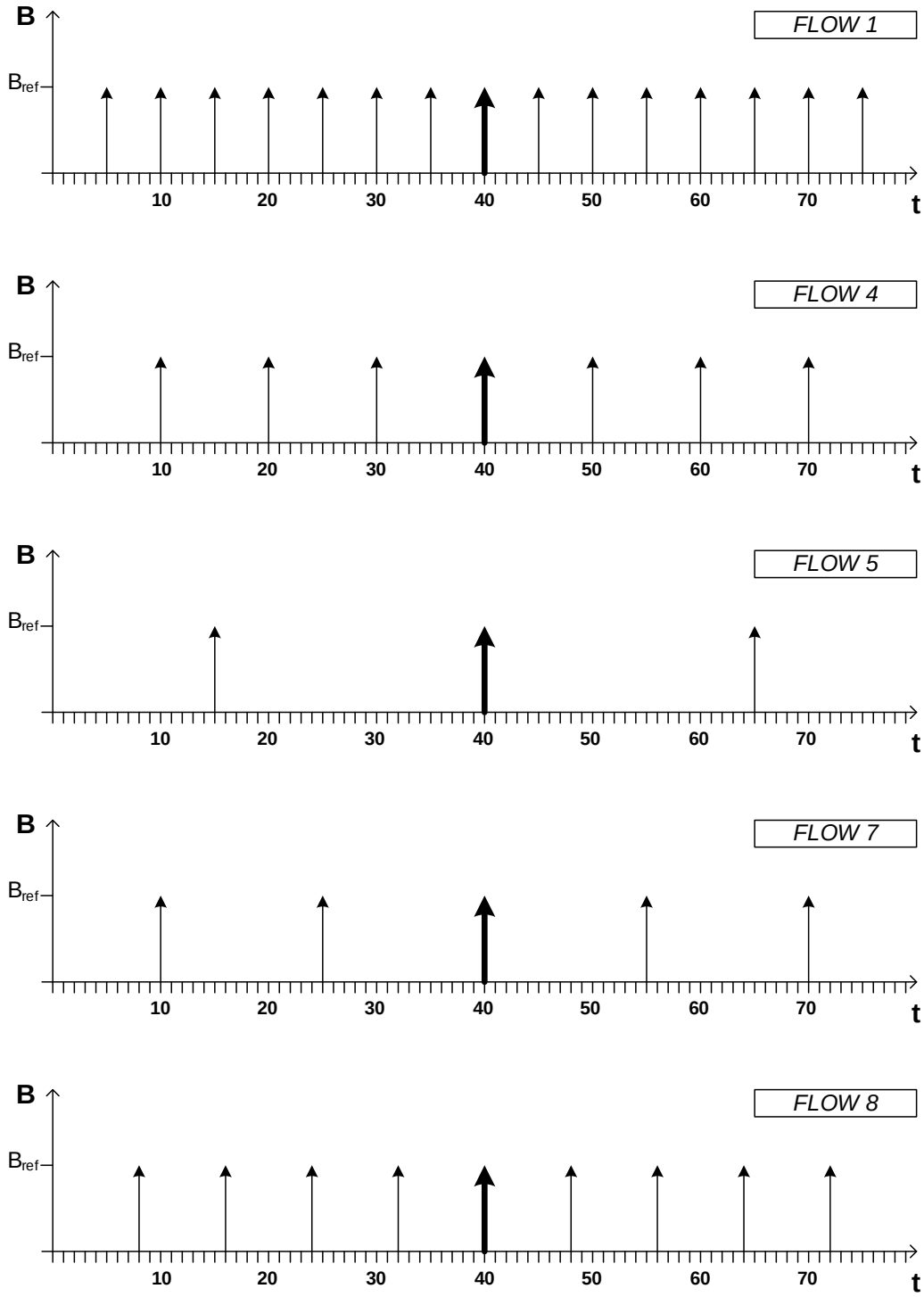


Figure 2.7: Flows Modelization

Each flow, as illustrated throughout the admission control modelization in the beginning of this chapter, is seen as a periodic discrete sequence of Kronecker deltas with amplitude B_{ref}. Figure 2.7 shows the trend of the maximum resource requirement found in the example detailed above, that is the sequence of flows 1-4-5-7-8. Starting times and periods are chosen randomly, in such a way that can be shown the simultaneous overlap of requests in a precise point in time, that in the current example is placed at unit time 40.

Obviously, optimizations to the full exploration of the tree of solutions are possible in order to further reduce the computational load of the algorithm. For instance, based on the matrix of intersections, the branches of solutions to be explored could be ordered in descending order according to their potential maximum value and thus, the exploration could be finished when a solution is found without requiring the full exploration of the tree of solutions.

Another optimization alternative could be to model bandwidth requirements B_i as multiples of an arbritrarily chosen one B_{ref}. In this case, a single flow larger than B_{ref} would be modeled as B_i/B_{ref} flows and the algorithm would not have to take into account the actual bandwidth requirement requests value, since they would be normalized to B_{ref} but just the total number of intersections in order to find the maximum.

While regarding the case of variable bandwidth requirements B_i, the algorithm has been modified. In fact the maximum resource requirement is no longer the maximum overlap of requests, but the 'heaviest' branch, that is the sequence containing the most consistent bandwidth request. It means that, for example, the intersection of 3 flows can require more resources with respect to a set of intersections composed by 5 flows.

The mechanism used to build the tree is the same as the one detailed above, while the procedure implemented to detect the maximum resource reservation is different; each flow is assigned a weigth proportional to its bandwidth request so that the 'heaviest' branch, as mentioned above, results from the sum of the weights involved in each branch.

2.5 Algorithms Performance Comparison

For validating and evaluating the performance differences between the *Worst Case*, *Heuristic*, *Diophantine* and *E-Diophantine* approaches, the respective algorithms have been implemented in matlab and the following experiment has been performed, which results are summarized in Figures 2.9 and 2.10. It is considered a system with 10 to 100 flows where for each one t_i and T_i are randomly chosen from a uniform distribution. The range of the uniform distribution is chosen depending on the granularity considered: 1 to 100 for granularity 1, 1 to 20 then multiplied by 5 for granularity 5 and 1 to 10 then multiplied by 10 for granularity 10. For illustration purposes B_i is taken as 1 in all cases.

Figure 2.9 shows the difference between the estimated maximum number of resources required by each of the approaches. As it can be observed, the difference increases as the number of flows increases as well as when a larger granularity is considered for all approaches but the *Diophantine* and *E-Diophantine*. Taking the *Diophantine* value as reference since it represents the exact solution, as expected the *Worst case* solution is the one presenting the largest differences to the actual values reaching differences of above 300%. Such large difference with respect to the actual requirements used would obviously result in a much lower usage of the network by services with QoS requirements than possible and thus, in a lower potential revenue for a network operator. In the *Heuristic* case, the larger the granularity the larger the difference to the actual value due to a limitation in the maximum LCM value that can be considered in a real implementation (10^7 in this system). Even worse, the estimation is below the actual value and therefore, its usage for admission control purposes could compromise the QoS guarantees in a network. On the other hand, the *E-Diophantine* estimation is always equal to the *Diophantine* one and thus, it confirms the correctness of Theorems 1 and 2.

In Figure 2.10 the corresponding differences in computational load are shown with respect to the *Heuristic* approach which is taken here as reference due to its implementation simplicity. The *Worst Case* is not considered since its computational load is obviously negligible but, as shown in Figure 2.9, its estimation of the actual resources used would also result in a much lower usage of the network by services with QoS requirements. It can be observed that the *Diophantine* solution, although exact, exceeds by far the computational load of the alternative solutions

considered and thus, it would not be feasible in practice [5]. For the lowest granularity considered, the *Heuristic* approach clearly outperforms in computational time the *E-Diophantine* solution with no loss of accuracy. However, as the granularity considered increases, the *E-Diophantine* performance is, in most of the cases, around three orders of magnitude faster than the *Heuristic* and at the same time always obtaining an exact estimation of the maximum requirement to be expected while the *Heuristic*, specially for granularity 1, clearly underestimates. Based on these results this work will focus only in the analysis of the proposed *E-Diophantine* solution.

2.6 Non-Ideal Conditions

Until this point have been considered that the data interarrival time and the resource requirements defined when requesting admittance in the system will be kept constant during the lifetime of a flow. However, in a real network, data arrival might be advanced due to aggregation of frames or delayed due to competition with other traffic for network resources in the path to a Base Station. Additionally, the resource requirements requested might vary due to changes of interference, channel conditions or movement of the mobile station.

In this section is evaluated the impact on the *E-Diophantine* maximum resource requirements and computational load when considering both aforementioned effects. In the case of advanced/delayed arrival three different jitter cases are considered: *No Jitter*, *Min Jitter* and *Max Jitter*. *Min Jitter* is defined as an arrival time which can vary in plus/minus one granularity unit with respect to the expected one. *Max Jitter* is defined as an arrival time which can vary in plus/minus one or two granularity units with respect to the expected one. With respect to the resource requirement variations, three different scenarios are considered depending on the percentage of flows using a specific modulation and coding scheme: Uniform, Capacity and Range. Table 2.2 details the percentages of flows considered to be using each of three modulation and coding schemes. In the experiment 16QAM 1/2 is considered as the reference unitary unit and accordingly, the resource requirement for B are 2/3 for 64QAM 1/2, 1 for 16QAM 1/2 and 2 for QPSK 1/2.

[5] For instance, in the 30 flows case the computation time in a 2*Quad Core simulation server took >1000 seconds.

	64QAM 1/2	**16QAM 1/2**	**QPSK 1/2**
Uniform	34%	33%	33%
Capacity	60%	30%	10%
Range	10%	30%	60%

Table 2.2: MCS distribution for the resource requirement experiment

Figure 2.11 shows the results for the three different jitter cases where it can be observed that, as expected, the larger the possible jitter variance considered the larger the maximum resource requirement estimations since the *E-Diophantine* algorithm has to consider the possibility of the resources request arriving at multiple times. In this case, this can be considered by the *E-Diophantine* algorithm by simply reducing the granularity of the starting time according to the jitter considered and thus, the computational time increase is negligible.

With respect to the consideration of different MCS distributions, Figure 2.12 shows the increase or decrease in the maximum bandwidth requirement estimation with respect to the *Uniform* distribution. The *E-Diophantine* solution can be easily configured to consider different distributions since it only requires to provide the input of B to the algorithm taking into consideration the corresponding distributions. As in the jitter case, since the modeling does not need to consider additional flows but just a different weighting for B the computational load increase is not noticeable.

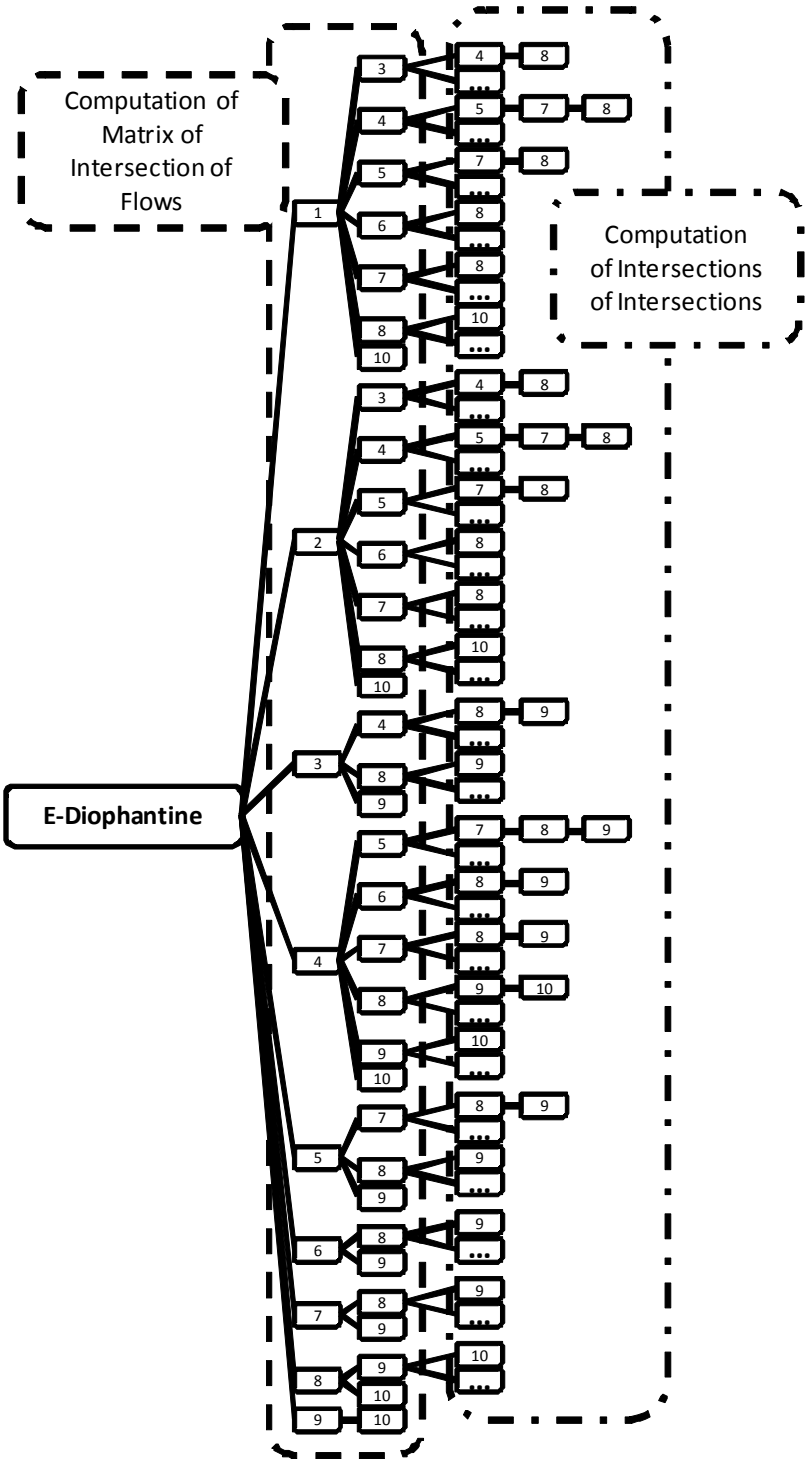


Figure 2.8: Tree of solutions of intersections for the E-Diophantine algorithm

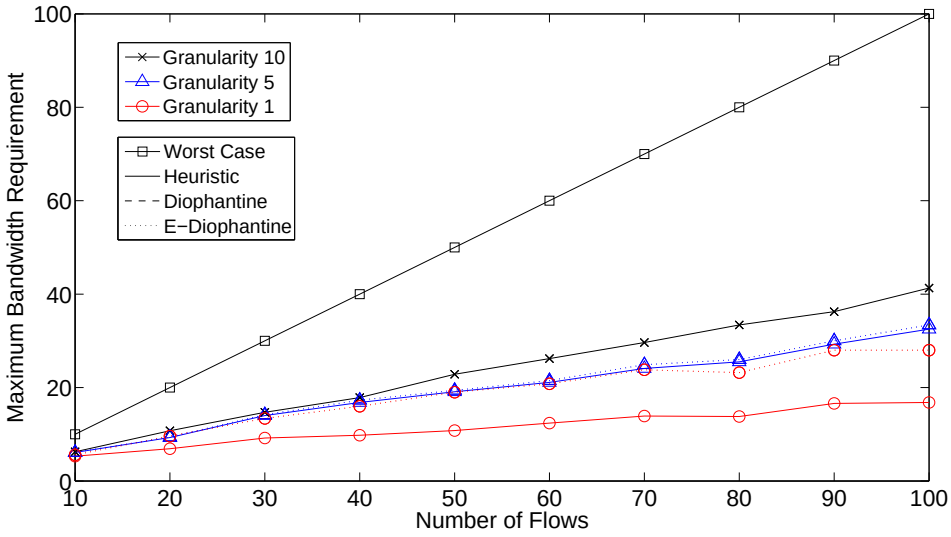


Figure 2.9: Expected maximum resources requirement

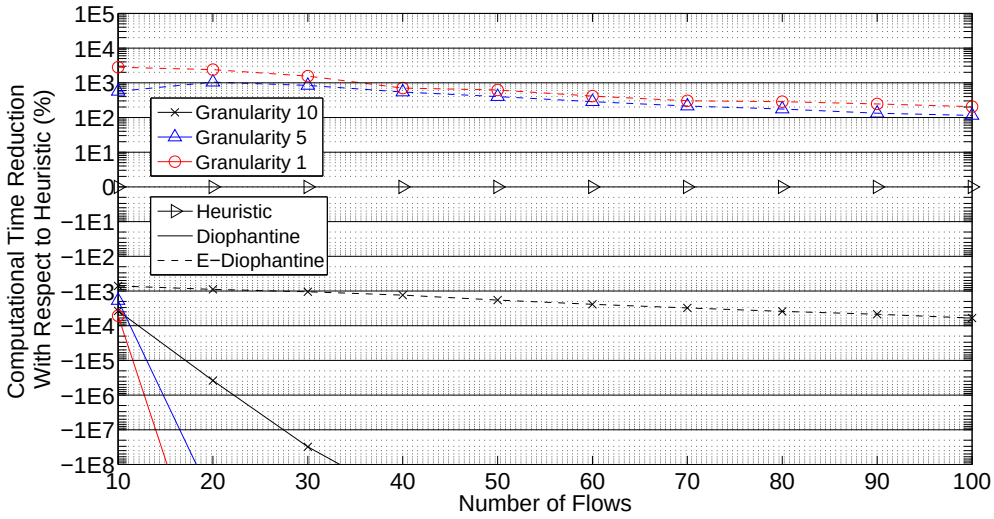


Figure 2.10: Computational time reduction with respect to Heuristic

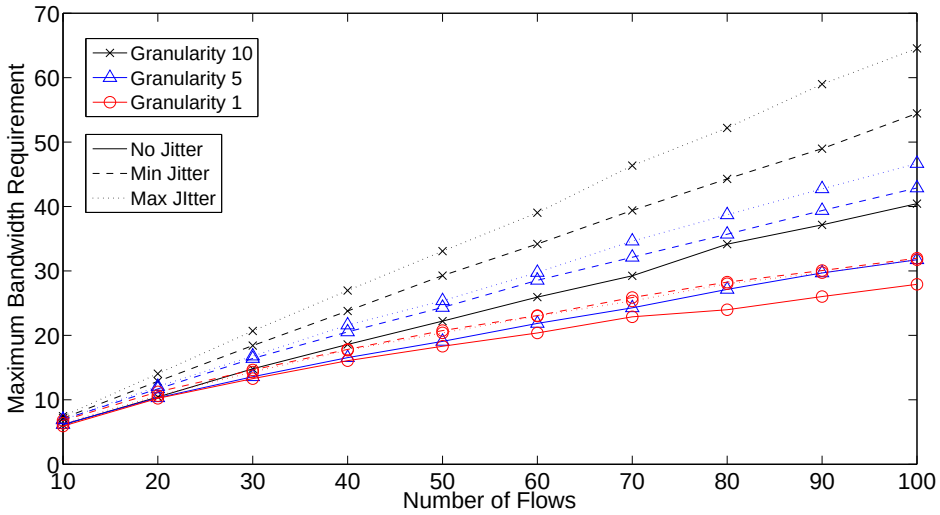


Figure 2.11: *E-Diophantine* performance considering Jitter

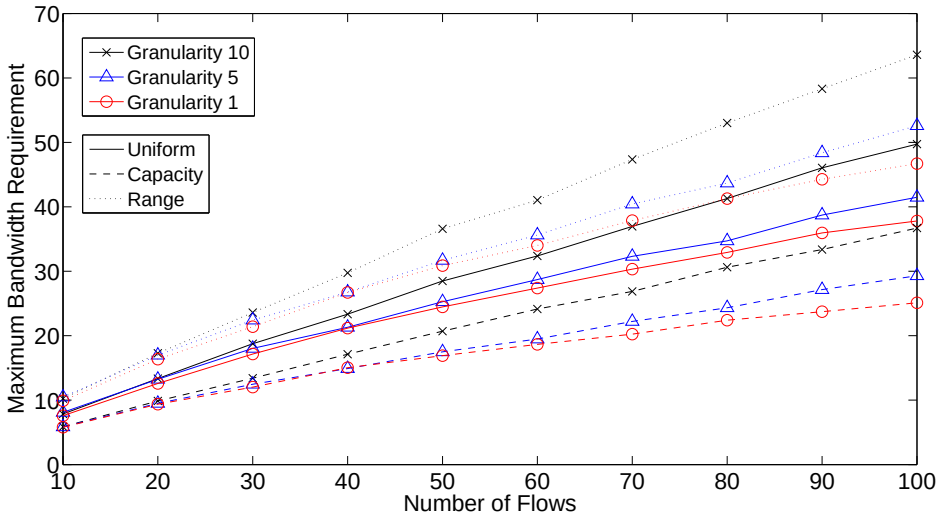


Figure 2.12: *E-Diophantine* performance considering MCS variations

Chapter 3

Performance Evaluation

In the previous section the performance of the proposed *E-Diophantine* solution as compared to its alternatives have been analyzed considering a generic scenario. In this section, the evaluation is completed by using OPNET's WiMAX simulator [10] to consider additional elements in the performance comparison that could have an impact in the maximum resource requirement estimation of the different approaches.

Examples of these elements are: wireless physical channel, Transport layer, Network layer, MAC layer, control plane signaling, realistic applications, QoS schedulers, number of subscriber stations, and so forth.

Then, in order to analyze the reliability of the algorithm projected for the realization of an efficient admission control mechanism, a realistic scenario has been created, whose specifications are detailed in the following.

3.1 Scenario

A scenario is setup according to Figure (snapshott to insert) and consisting of one Base Station (BS) and five Subscriber Stations (SS) where each station is configured to send and receive traffic from their corresponding pair in the wired domain of its type of application, i.e., one station sends and receives Voice traffic (without silence suppression), a second station sends and receives Voice traffic (with silence suppression), a third one receives a Video stream, a fourth one does an FTP download and the last one does Web browsing.

Then, the number of stations is increased in multiples of five stations up to 125 in total, always keeping the relation of 1/5 of stations of each application type. The QoS scheduling policy chosen is Strict Priority applied first to fulfill the Minimum Reserved Traffic Rates (MRTRs) and then, the Maximum Sustained Traffic Rates (MSTRs).

The length of the simulations performed is 120 seconds with a warm-up phase of 10 seconds. The number of seeds used to obtain average throughput values has been increased until their 95% confidence intervals did not overlap. In the case of the delay performance metric, the values represent the 95% percentile of the delay (CDF95) considering all simulation runs.

The configuration used for the different applications is detailed below:

- *Voice* **G.711 Voice codec**
  - Data rate: 64kb/s.
  - Frame length: 20ms.
  - Mapped to *UGS* in the DL (BS → SS) and UL direction (BS ← SS).
- *Voice (silence suppression)* **G.711 Voice codec**
  - Data rate: 64kb/s.
  - Frame length: 20ms.
  - Talk spurt exponential with mean 0.35 seconds.
  - Silence spurt exponential with mean 0.65 seconds.
  - Mapped to *ERT-VR* in the DL and to *ertPS* in the UL[1].
- *Video* **MPEG-4 real traces** [11]
  - Target rate: 450 kb/s.
  - Peak: 4.6 Mb/s.
  - Frame generation interval: 33ms.
  - Mapped to *RT-VR* in the DL and to *rtPS* in the UL.

[1]Note that the ERT-VR, RT-VR and NRT-VR data delivery services in the donwlink direction correspond to ertPS, rtPS and nrtPS in the uplink

- *FTP*
  - Download of a 20MB file.
  - Mapped to *NRT-VR* in the DL and to *nrtPS* in the UL.
- *Web Browsing*
  - Page interarrival time exponentially distributed with mean 60s.
  - Page size 10KB plus 20 to 80 objects of a size uniformly distributed between 5KB and 10KB [12].
  - Mapped to the *BE* service both in the DL and UL direction[2].

Tables 3.1, 3.2 and 3.3 summarize the parameters utilized for the performance evaluation in both the directions, downlink and uplink.

WiMAX PHY Layer Config.	
Base Freq. (GHz)	2.5
Bandwidth (MHz)	10
Frame Duration (ms)	5
Symbol Duration (μs)	102.86
Number of Subcarriers	1024
DL Subfr. # Symbols	35
UL Subfr. # Symbols	12
DL Subfr. # Subch.	30
UL Subfr. # Subch.	35
# Data Subc./Subch	24
# SSs 64 QAM (3/4)	60%
# SSs 16 QAM (3/4)	30%
# SSs QPSK (1/2)	10%

Table 3.1: Performance Evaluation Parameters - Physical Layer

[2]http://www.websiteoptimization.com/speed/tweak/average-web-page/

Data Delivery Services	
UGS	MRTR: 80 Kb/s
	Max. Lat: 20 ms
ERT-VR	MRTR: 80 Kb/s
	Max. Lat: 20 ms
RT-VR	MSTR: 2 Mb/s
	MRTR: 500 Kb/s
	Max. Lat: 33 ms

E-Diophantine	
UGS	B_{UGS}: 1600 bits
	T_{UGS}: 20 ms
ERT-VR	B_{ERT}: 1600 bits
	T_{ERT}: 20 ms
RT-VR	B_{RT}: 16500 bits
	T_{RT}: 33 ms

Table 3.2: Performance Evaluation Parameters - DL Scheduling Services

Data Delivery Services	
UGS	MRTR: 80 Kb/s
	Max. Lat: 20 ms
ertPS	MRTR: 80 Kb/s
	Max. Lat: 20 ms
rtPS	MSTR: 1.12 Kb/s
	MRTR: 1.12 Kb/s
	Max. Lat: 1 s

E-Diophantine	
UGS	B_{UGS}: 1600 bits
	T_{UGS}: 20 ms
ertPS	B_{ERT}: 1600 bits
	T_{ERT}: 20 ms
rtPS	B_{RT}: 1120 bits
	T_{RT}: 1 s

Table 3.3: Performance Evaluation Parameters - UL Scheduling Services

3.2 Performance Results

As previously mentioned, in the current section is introduced and analyzed the graphics related to the performance issues of the system realized with OPNET's WiMAX simulator. In order to prove results' reliability and coherence have been collected delay and throughput statistics, whose trends confirm the assumptions made for the scenario's specifications and the consistency of the AC algorithm respectively.

3.2.1 Downlink Throughput

As detailed in Figure 3.1, assuming that C_{OH} and C_R require in the best case 3 symbols, the maximum bandwidth available for data communication calculated for the remaining 30 symbols (C_D = 15 PUSC slots) is approximately 21 Mb/s[3] considering a 64QAM 3/4 modulation and coding scheme for all bursts. Hence the system begins to reject connections when, as shown in Figure 3.1, the CAC line, that represents the maximum bandwith request prediction, exceeds the subframe capacity.

In the graphics is shown the peak and average throughput experienced in the downlink by the different application types as compared to the peak capacity estimations of the different approaches described in the previous sections. The throughput of the different applications is aggregated according to whether it is considered for admission control (*Premium* traffic: UGS+ERT-VR+RT-VR), or not (*Regular* traffic: NRT-VR+BE). Additionally, the average throughput of each single data delivery service belonging to the *Premium* group is provided as a reference. From the performance results in Fig.3.1 the first remarkable result is that the peak of *Premium* traffic is in some cases above the peak estimated with the different admission control algorithms considered but the *Worst Case* one. The reason for this result is the 2Mb/s MSTR configured for RT-VR which allows video applications to get more than its 500 Kb/s MRTR if there is leftover capacity after serving all MRTRs. Note that the *Worst Case* estimation is too conservative and therefore, it will not be considered in the reminder of this section.

As the number of stations increases, the difference between the admission control estimations and the throughput peak of *Premium* traffic decreases. Note that the larger the amount of *Premium* traffic in the network, the lower the opportunities to go above the MRTR value. Eventually a point is reached where even the MRTR guarantees can not be satisfied, see crossing point between 20 and 25 stations per data delivery service. Moreover, as the number of flows in the system increases, the signaling overhead required for the DL-MAP increases as well, resulting in a lower *Premium* average throughput. For illustration purposes, an additional *E-diophantine* case has been added, *E-Dioph (1Mbps)*, where the MRTR for RT-VR

[3](35 symbols-3(preamble+maps overhead))*30 subchannels*24 data subcarriers/subchannel *6 bits/symbol *3/4(redundancy)/5ms(frame duration)=20.736 Mb/s

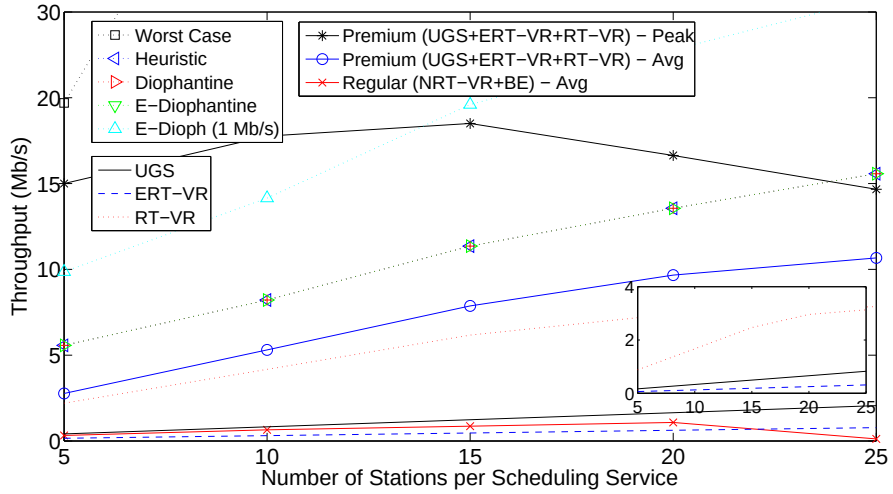


Figure 3.1: Downlink Throughput

has been configured to 1 Mb/s instead of 500 Kb/s. This case provides an example of how the admission control estimation would vary by allowing bursty traffic to transmit significantly above their average.

3.2.2 Downlink Delay

With respect to the delay performance, the results are shown in Fig. 3.2. As expected, when the wireless resources become scarce, the delay experience degrades according to the traffic priority. In the case of RT-VR traffic, in contrast to UGS and ERT-VR, the delay experienced increases constantly. This is due to the performance metric chosen, 95% percentile of the delay (CDF95), which yields a close to worst case delay for each application traffic and thus, as the number of flows grows, it increasingly represents the Video peaks that can not be absorbed because there is not enough remaining capacity after serving all MRTRs.

The delay performance of BE, which increases very rapidly, is due to the simple QoS scheduling policy used, Strict Priority, resulting in BE traffic being served only if the rest of the available traffic has already been served. Other QoS scheduling policies friendlier to low priority traffic, e.g., Weighted Round Robin, could significantly improve the BE performance with a negligible impact on higher traffic

priority classes.

Finally, both the NRT-VR and BE delay performance experience an extreme degradation after the 20 stations per data delivery service point. Note that this is where the estimation of the different admission control algorithms but the *Worst Case* crosses the *Premium* peak throughput and therefore, the probability for NRT-VR and BE traffic to be served decreases significantly.

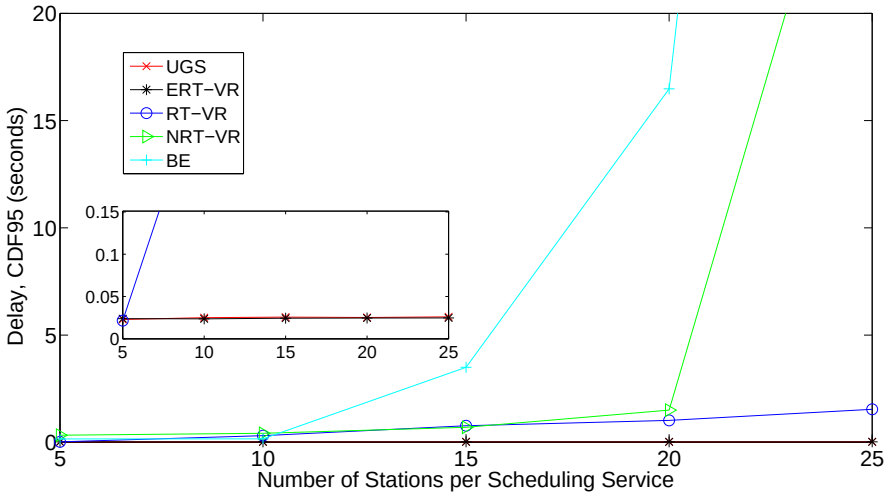


Figure 3.2: Downlink Delay

3.2.3 Uplink Throughput

As depicted in Figure 3.3, the uplink throughput increases linearly.

Depending on the MCS percentages distribution, as shown in Table 2.2, and from the parameters of Tables 3.1 and 3.2, the effective data capacity of the uplink subframe is about 9 Mb/s[4] and, as depicted in the graphics, neither the CAC line or the peak rates exceed that value.

There is no *Premium* application generating data above the agreed MRTR and therefore, the *Premium* Peak throughput is always around the predicted one.

[4] 12 symbols*(35 subchannels-1(ranging overhead))*24 data subcarriers/subchannel *6 bits/symbol *3/4(redundancy)/5ms(frame duration)=8.812 Mb/s

As shown in Fig.3.3, the average throughput of all traffic types increases linearly[5], which indicates that the capacity in the uplink is sufficient to serve all traffic needs, resulting in a non-degraded service arrangement.

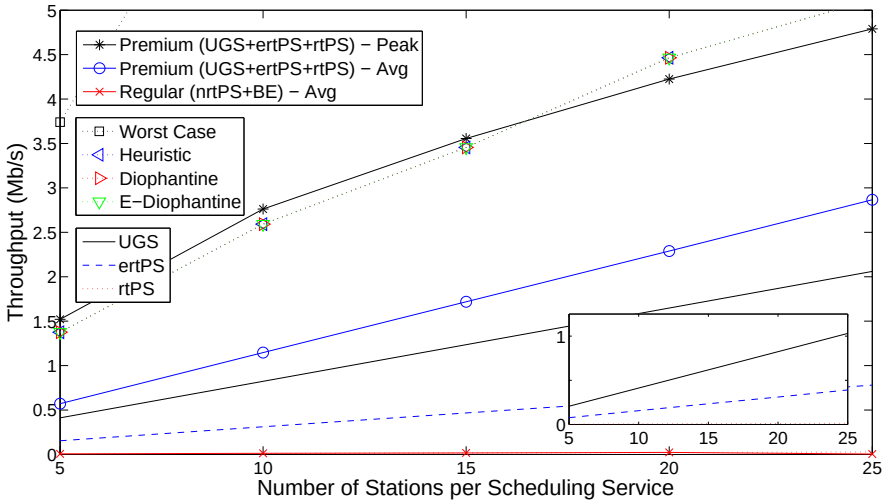


Figure 3.3: Uplink Throughput

3.2.4 Uplink Delay

As depicted in Figure 3.4, due to the bursty nature of BE and the QoS scheduler employed, BE takes preference over nrtPS when active since the counter for its share of the remaining capacity is zero while the one for nrtPS is positive. For this reason, when the number of stations increases, nrtPS connections present minor delay; extra bandwidth decreases, and less contention is experienced.

While regarding *Premium* traffic, since there is no system saturation, as depicted in Figure 3.3, those lines in the graphics are nearly constant, around 0.

[5]Note that both the FTP and Web applications only generate TCP Acks in the uplink and therefore its average throughput is close to zero

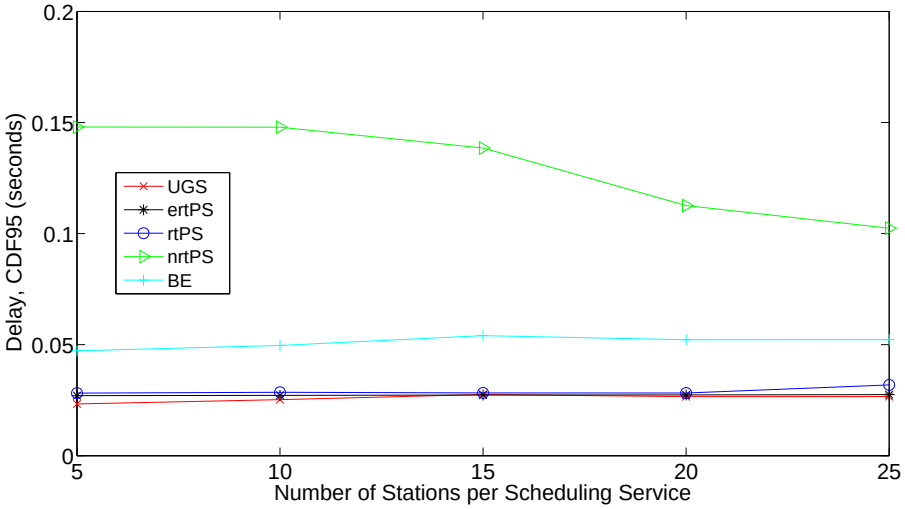


Figure 3.4: Uplink Delay

3.3 Observations

Based on these results it can be concluded that the different solutions described in Chapter 2 but the *Worst Case* could be effectively used to predict whether a new reservation should be admitted in the system.

The selection of which algorithm would be more appropriate for a specific case should be taken considering the results presented in Section 2.5, where is analyzed the efficiency of each admission control proposal in terms of accuracy and computational load, hence the possibility to select the mechanism that fits better the network scenario according to the range of applications involved.

Chapter 4

Multi-Hop Relay Extension

Multi-Hop relay systems represent a potentially attractive option for extending coverage and increasing throughput of broadband wireless access networks. A relay-based approach can be pursued, wherein low cost relay stations (RSs) are introduced into the network to help extend the range, improve service, boost network capacity, and eliminate dead spots, all in a costeffective fashion [13]. In Figure 4.1 an example of a multi-hop relay scenario with an uplink and downlink communication is provided.

The IEEE 802.16j standard [14] specifies two different scheduling modes in order to arrange the bandwidth allocations for a Subscriber Station (SS) or a Relay Station (RS): *centralized* and *distributed*. In the former, the Multi-Hop Relay Base Station (MR-BS) determines the scheduling for all nodes in the system, conversely in the latter the bandwidth allocation of an RS's subordinate station can be determined by the RS itself. Furthermore the standard defines two different relay modes of operation: *transparent* and *non-transparent* [15].

- **Transparent Mode**: the RSs do not generate its own preamble and overhead mapping information, thus the SSs need to always be in range of the MR-BS enhancing the capacity within the basic coverage area; this type of relay is of lower complexity and only operates in a centralized scheduling mode for topology up to two hops. In Figure 4.2 is depicted the transparent mode frame structure while Figure 4.3 represents the scheme of a transparent relay mode communication, as detailed in [16].

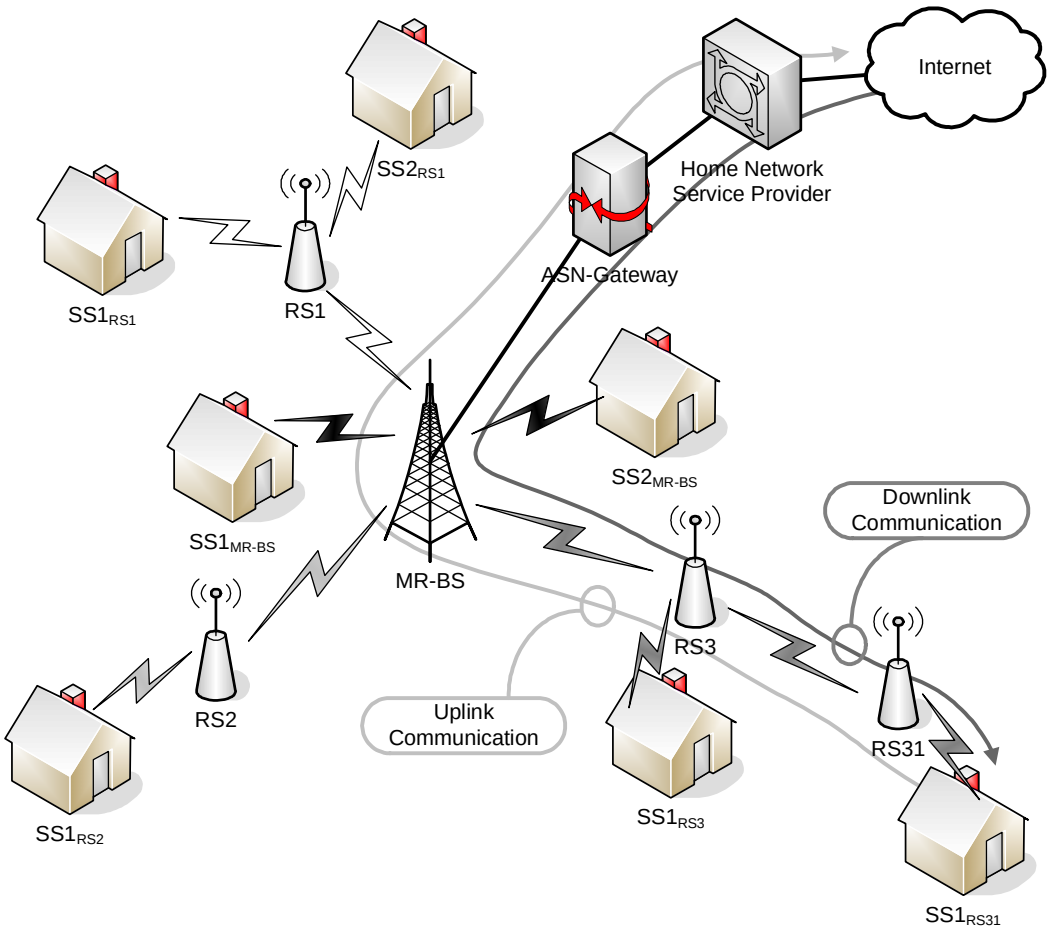


Figure 4.1: IEEE 802.16j Multi-Hop Relay Scenario Example

- **Non-Transparent Mode**: the RSs generate their own framing information providing increased coverage in the case of distributed scheduling, while if the relay mode is set on centralized scheduling they forward those information provided by the MR-BS. A functioning illustration is provided in Figure 4.5, whose details are presented in [16].

In this chapter is analyzed the case of non-transparent relay mode with distributed scheduling, as shown in Figures 4.5 and 4.6. The next sections investigate two important aspects of such systems; capacity modeling is discussed in Chapter 4.1, while in Chapter 4.2 is evaluated the admission control proposed solution.

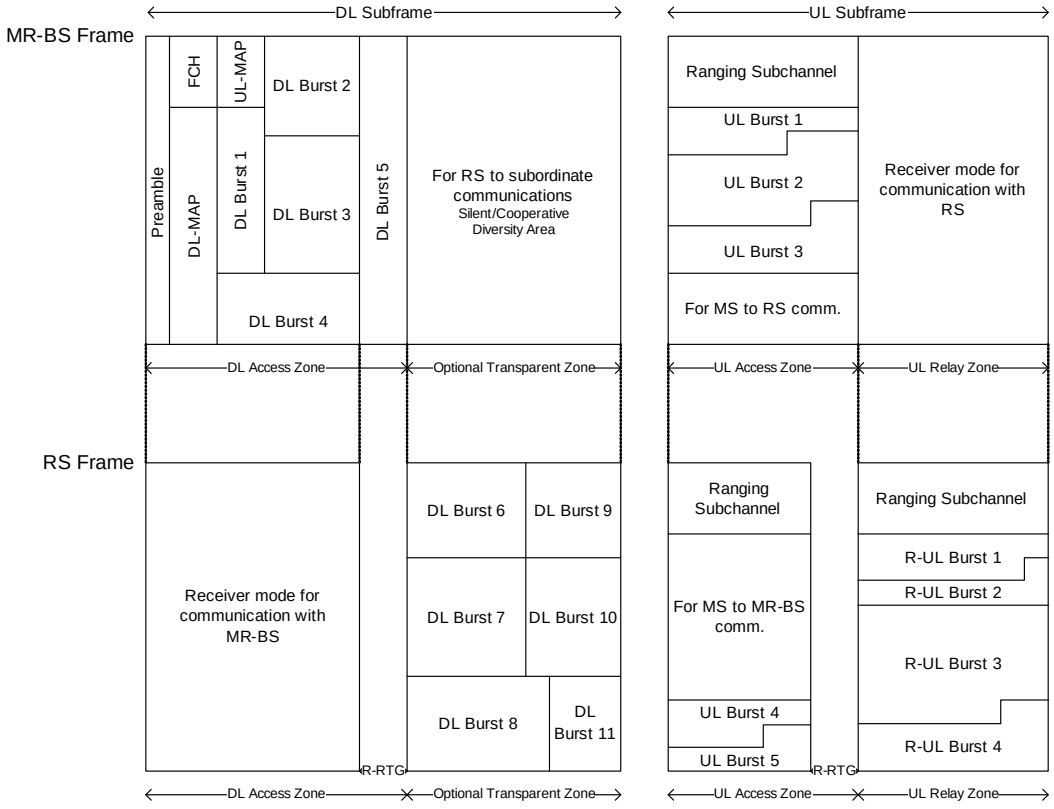


Figure 4.2: Transparent Mode frame structure

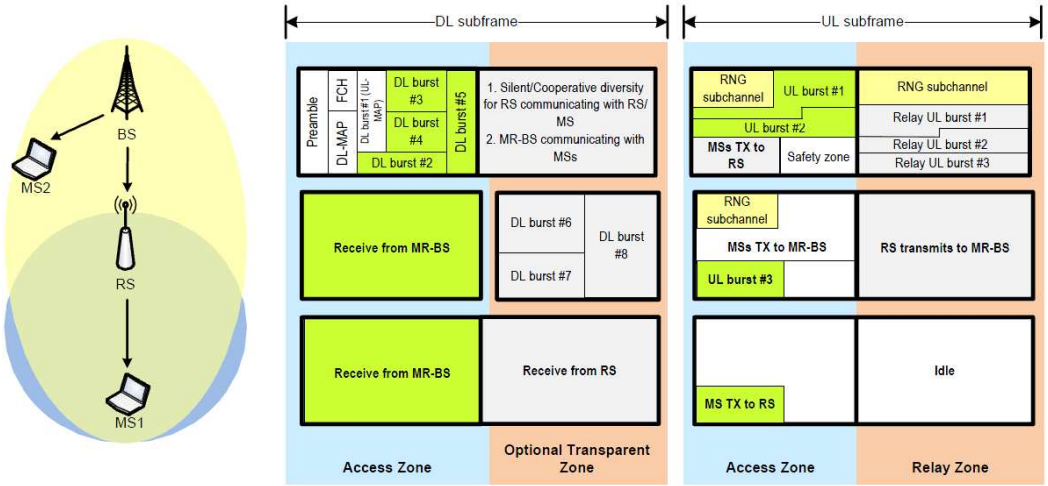


Figure 4.3: Transparent Relay

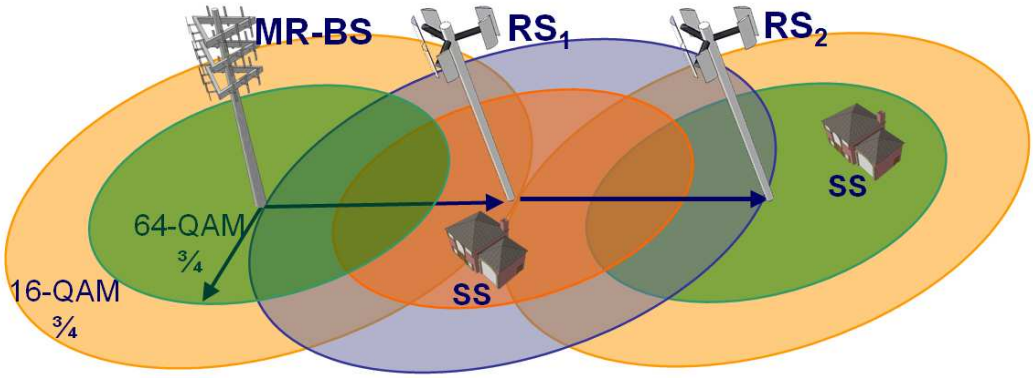


Figure 4.4: Increased coverage in non-transaparent relay mode scenario

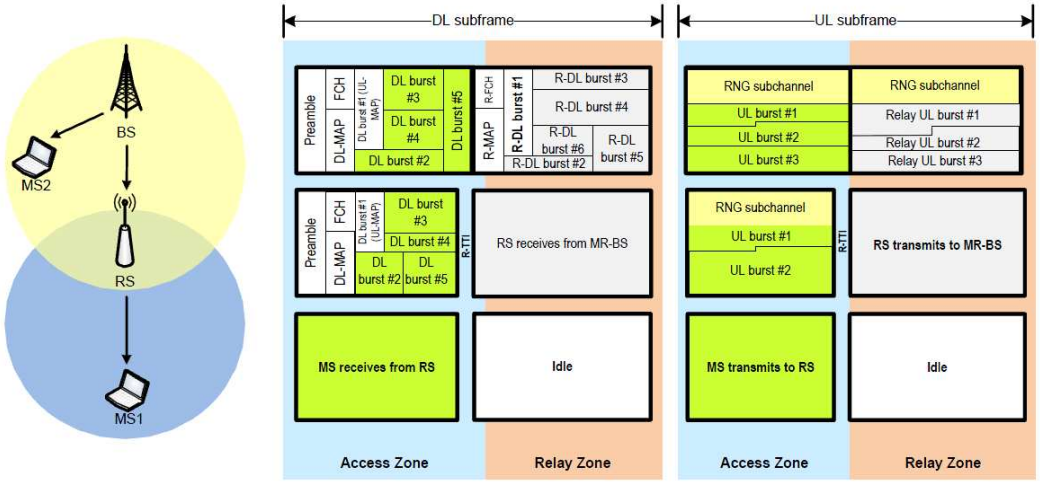


Figure 4.5: Non-Transparent Relay

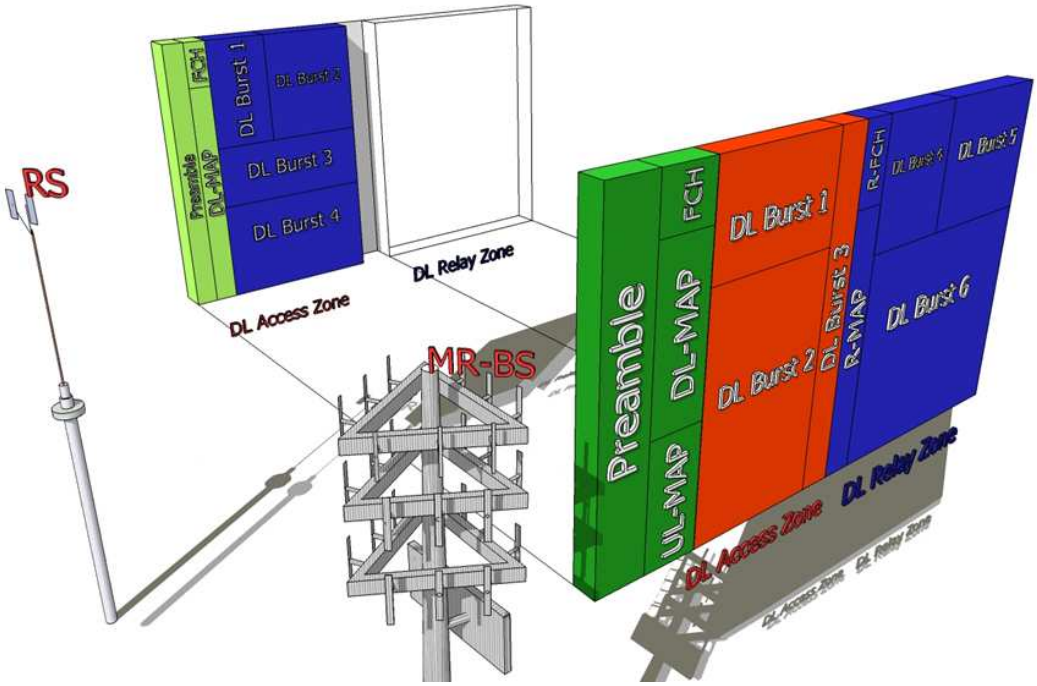


Figure 4.6: Non-Transparent Mode Scenario

4.1 Capacity Modeling

Figures 4.7 and 4.8 depict the capacity modeling of a system operating with a non-transparent mode relay fashion, relatively to the downlink and the uplink respectively. Both the DL and UL subframes for the MR-BS and the RSs show an access zone where takes place the communication to or from the SSs, and a relay zone that represents the communication between the MR-BS and the RS_i, or between the RS_i and the RS_{ij}, in both the direction. RS_i and RS_{ij} represent the aggregation of all the RSs directly communicating with the MR-BS (first hop) and the aggregation of all the RSs communicating with the RS_i (second hop) respectively. For simplicity a system deployed in one dimension is assumed, as depicted in Figure 4.4. As described in 1.1 the capacity of each active subframe is basically composed in this model of three parts: one including the diverse framing information, one representing the available slots for data communication, and one reserved part for potential retrasmission, MCS changes, or transmission of best effort traffic.

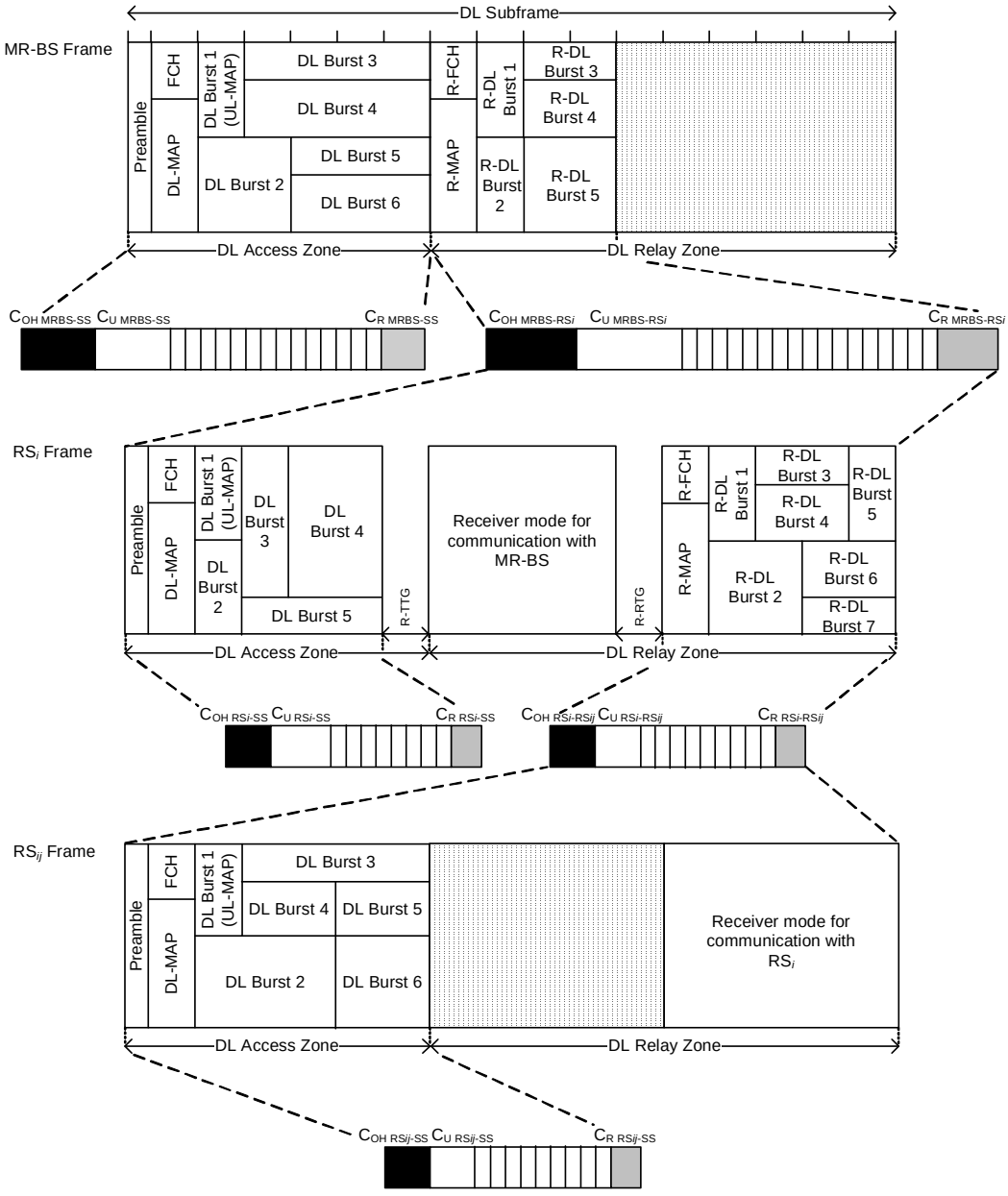


Figure 4.7: Multi-Hop Capacity Modeling - Downlink

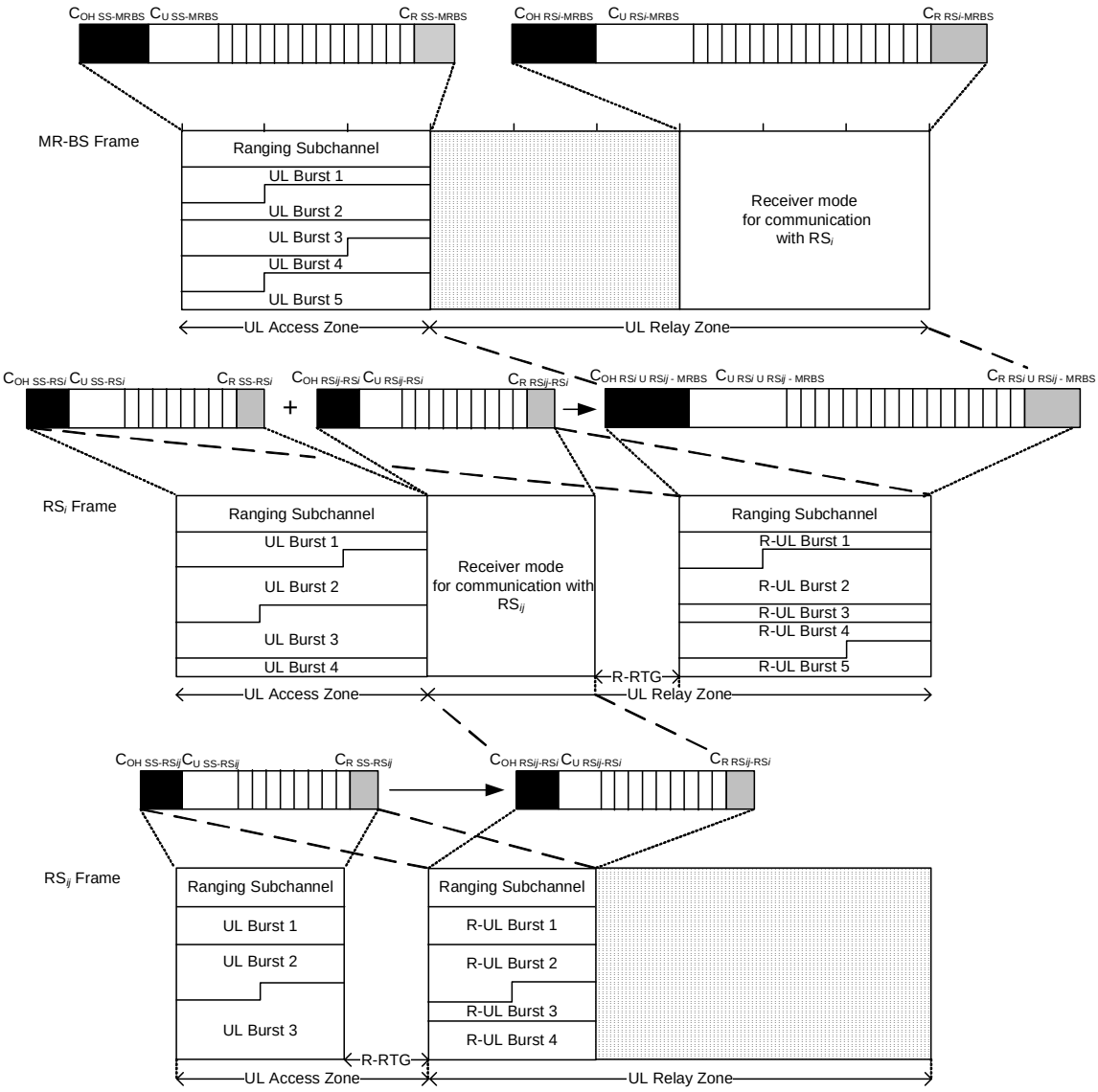


Figure 4.8: Multi-Hop Capacity Modeling - Uplink

4.2 Admission Control Algorithm

The $E - Diophantine$ algorithm presented in Chapter 2.4 can be extended to be applicable to the IEEE 802.16j standard for multi-hop relay scenarios. In the following is described in detail the proposed $E - Diophantine$ multi-hop relay extension.

In the IEEE 802.16j multi-hop relay case, incoming flows from a MR-BS or RSs to the next RS can be considered by the $E-Diophantine$ solution as just additional flows with QoS requirements. As such, the $E - Diophantine$ solution itself does not need any extension but the admission control algorithm using it. When a new flow requests admittance in the system the admission control algorithm should determine if the destination, in the case of a downlink request, or the source, in the case of an uplink request, are associated to an RS and in such a case consider it for the maximum capacity requirement computation in the following way.

First, two cases need to be differentiated. If no RS is involved for the new flow request, the $E - Diophantine$ solution described in 2.4 can be directly applied. On the other hand, if a RS is involved, the increase in the maximum capacity requirement needs to be checked for the Base Station and Relay station/s involved in the flow path until its destination. In the latter case, starting from the first Base Station or RS in the new flow data path and ending at the last MR-BS or RS within a local WiMAX network, the new maximum capacity requirement will be computed sequentially and if at any step it is considered to be above the maximum capacity available, the request will be rejected. In order to compute the new maximum capacity requirement at each MR-BS and RS/s involved, the set of flows already accepted in the system plus the new one need to be considered, taking into account that the arrival of the flow to each next MR-BS or RS will be increased by an integer number of WiMAX frames duration, N_f, according to the processing capabilities of the MR-BS and RSs.

4.2.1 Downlink

In the downlink case, considering a set of flows with QoS requirements coming from the WiMAX Core Network with their reservations defined as follows $R_{CN} = [r_{CN1}, r_{CN2}, ..., r_{CN_N}]$, for each subsequent RS the set of flows to be considered by each Relay in the flow path until its destination, R_{RS_M}, including the new flow

requesting admittance can be expressed as

$$R_{RS_M} \subseteq R_{RS_{M-1}} \subseteq R_{RS_{M-2}} ... \subseteq R_{CN} \tag{4.1}$$

and the periodic bandwidth requests for any reservation $j \in R$ at Relay $R_i \in i = 1..M$ as

$$B_j \cdot \delta(t_j + i \cdot N_f + n_j \cdot T_j) \tag{4.2}$$

Thus, the $E - Diophantine$ solution can be applied to obtain the maximum expected resource requirement at each RS by considering the corresponding subset of reservations of R_{CN} and increased starting time $t_j + i \cdot N_f$.

4.2.2 Uplink

In the uplink case, in contrast to the downlink one, at each hop from the source the number of reservations to be considered to find the maximum resource requirement might increase. Considering a set of flows with QoS requirements originated by the SSs associated with a Relay i with their reservations defined as follows $R_{SS_{Ri}} = [r_{SS1}, r_{SS2}, ..., r_{SSN}]$, for each subsequent RS the set of reservations to be considered by each Relay in the flow path until the MR-BS, R_{BS}, including the new flow requesting admittance can be expressed as

$$R_{BS} \supseteq R_{RS_M} \supseteq R_{RS_{M-1}} ... \supseteq R_{RS_1} \tag{4.3}$$

and the periodic bandwidth requests for any reservation $j \in R$ at Relay $R_i \in i = 1..M$ or MR-BS $i = M + 1$ as

$$B_j \cdot \delta(t_j + (i-1) \cdot N_f + n_j \cdot T_j) \tag{4.4}$$

Thus, similar to the downlink case, the $E - Diophantine$ solution can be applied to obtain the maximum expected resource requirement at each RS and the MR-BS by considering the corresponding subset of reservations of R_{BS} and increased starting time $t_j + (i-1) \cdot N_f$.

Chapter 5

Conclusion

Networks with QoS guarantees require an admission control algorithm able to estimate the increase in allocated capacity needed if a new resource reservation is admitted. In this work the *E-Diophantine* solution has been proposed, along with its mathematical foundations, and its benefits evaluated as compared to three alternative approaches, namely: *Worst Case*, *Heuristic* and *Diophantine*. The performance comparison comprised both accuracy and computational load analysis in a generic scenario as well as an evaluation using OPNET's WiMAX simulator in a realistic scenario.

The main conclusions that can be drawn from these results are:

- the *E-Diophantine* algorithm can be successfully used to predict the maximum allocated capacity demand of admitted QoS reservations in realistic scenarios;
- the simpler *Heuristic* approach can outperform the *E-Diophantine* one in computational terms if limitations in the period between resource allocations can be imposed;
- the larger the degree of flexibility allowed for defining the resource reservation periods, the larger the benefit of the *E-Diophantine* solution both in accuracy and computational load terms.
- the extension of the *E-Diophantine* algorithm can support admission control in multi-hop relay networks (IEEE 802.16j).

5.1 Acknowledgments

The research leading to these results received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement Nº 214994 [17].

Appendix A

Documentation

In this section is illustrated the organizational framework of the research carried out, the so-called *modus operandi*.

As shown in Figure A.1, the overall structure can be seen as a filter with an input connected to a two-steps inner process that provides the output. Each block will be detailed in the following, according to a folder-based scheme.

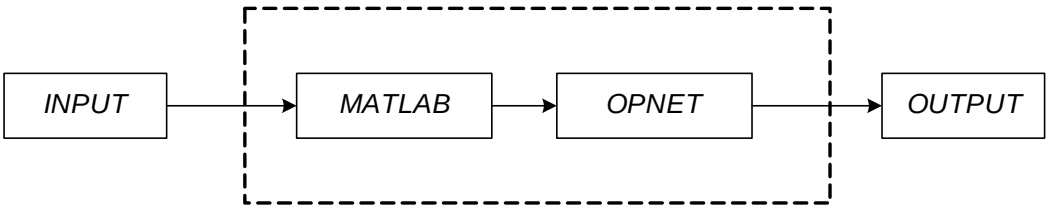


Figure A.1: Modus Operandi - Block Scheme

A.1 Input

The input block is given by the documentation used to understand the system functionalities detailed in the IEEE standards, 802.16-2009 and 802.16j, where are described the single mode and multi-hop relay mode respectively. Furthermore there is a folder containing the State of the Art relative to the admission control in

WiMAX networks, whose contents have been used to initiliaze the research.

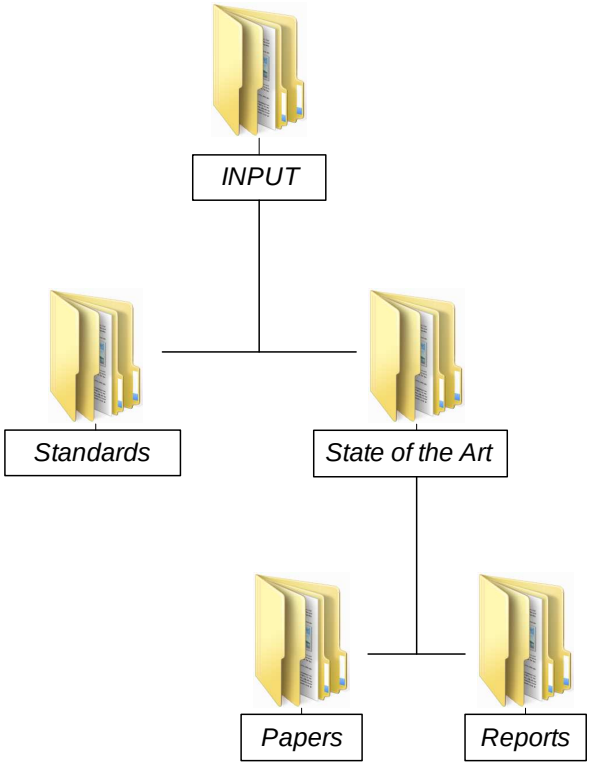


Figure A.2: Modus Operandi - INPUT

A.2 Matlab

The proposed solution presents its roots in matlab implementation of the admission control algorithms. The subfolder *CAC Algorithm* contains the codes generated to test and compare all the algorithms in order to evaluate individual reliability and efficiency (see Chapter 2.5), while in the subfolders *Process Statistics* and *Performance Evaluation* are, respectively, the codes used to process the statistics generated after OPNET simulations, and the codes executed in order to compare the proposed prediction mechanism with realistic issues coming from OPNET ad-hoc scenarios (see Chapter 3).

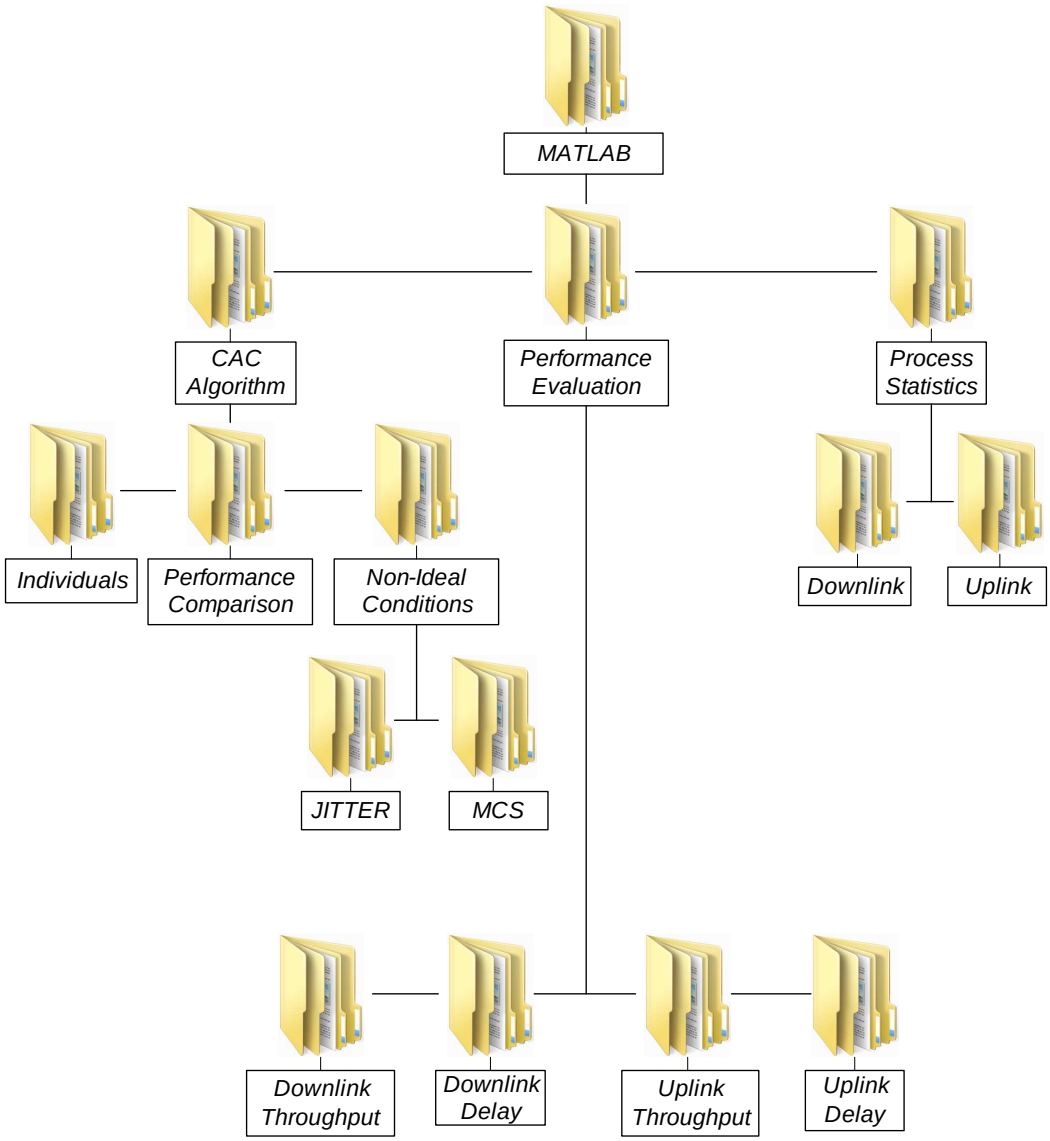


Figure A.3: Modus Operandi - MATLAB

A.3 Opnet

All the OPNET project files, with scenarios' topologies and applications included, are contained in the subfolder *Source Files*. The scripts used to collect the statistics coming from OPNET simulations are in the subfolder *Collection Scripts*, while all the collected statistics and the video scripts are contained in the subfolder *Trace Files*.

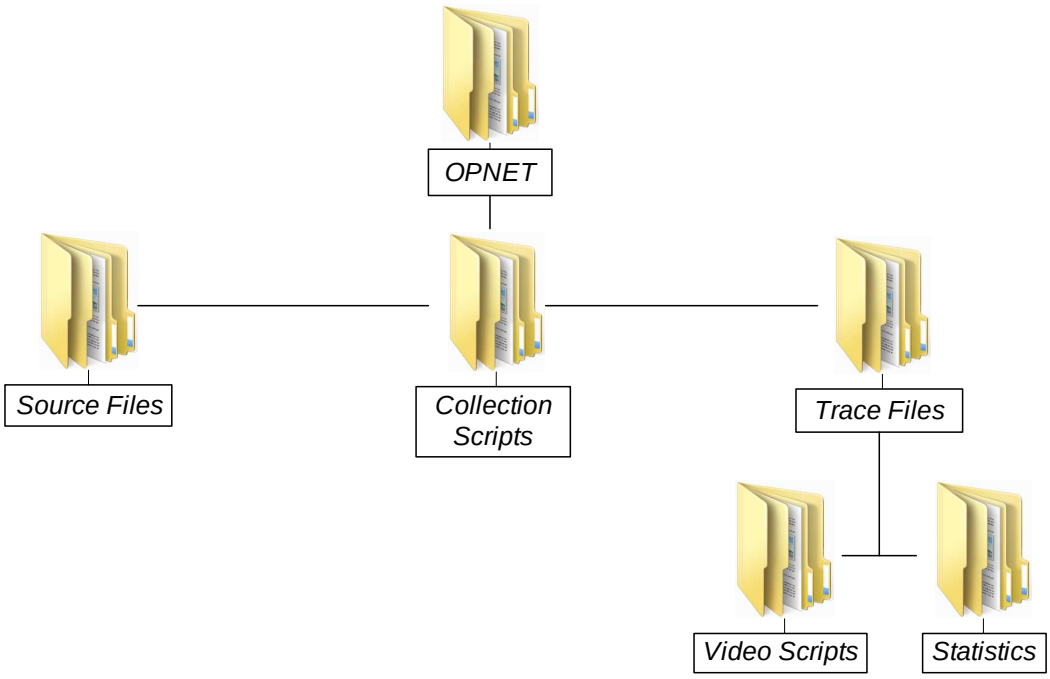


Figure A.4: Modus Operandi - OPNET

A.4 Output

Finally, all the results are collected and organized in the following documents:

- *Deliverable*: the research done led to the contribution to a chapter of the deliverable for the EU FP7 Project Carrier Grade Mesh Networks (CARMEN).
- *Paper*: the proposed solution has been developed in a paper accepted at the IEEE WCNC '10, 'E-Diophantine - An Admission Control Algorithm for WiMAX Networks'.

- *Patent*: *E-Diophantine* algorithm represents the central idea of a patent application, 'A Novel Probalistic Data Structure Supporting Wildcard, Cardinality and Threshold Queries'.
- *Master Thesis*: the current work.

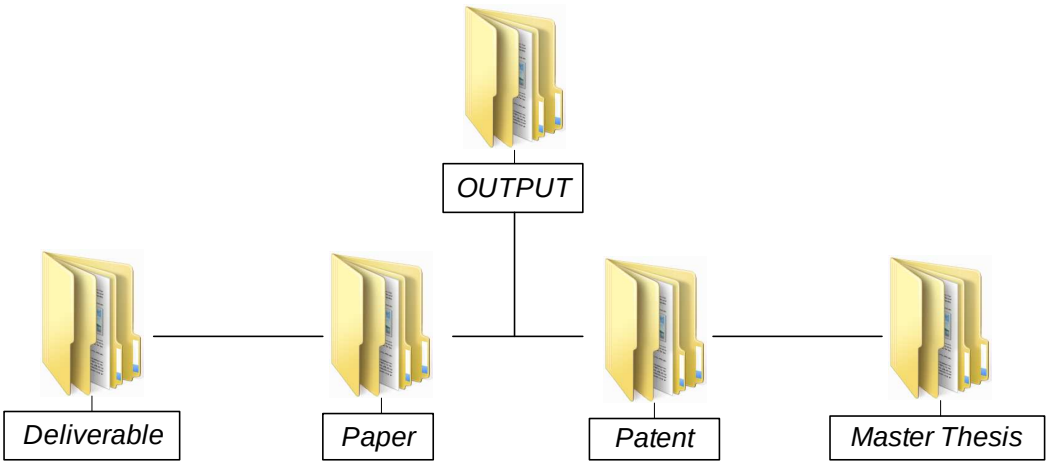


Figure A.5: Modus Operandi - OUTPUT

Appendix B

Abbreviations and acronyms

3G	Third Generation
ARQ	(Automatic Repeat Request)
ASN	Access Service Network
BE	Best Effort
BS	Base Station
BWA	Broadband Wireless Access
CID	Connection Identifier
CSN	Connectivity Service Network
DL	Downlink
DSA-REQ	Dynamic Service Addition REQuest
ERTPS	Extended Real Time Polling Service
FCH	Frame Control Header
FDD	Frequency Division Duplexing
FUSC	Full Usage of Subcarriers
HARQ	(Hybrid Automatic Repeat Request)

HCCA	HCF Controlled Channel Access
HCF	(Hybrid Coordinator Function)
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
LOS	Line-ff-Sight
MAC	Medium Access Control layer
MAN	Metropolitan Area Network
MCS	Modulation and Coding Scheme
MR-BS	Multi-Hop Relay Base Station
MRTR	Minimum Reserved Traffic Rate
MSTR	Maximum Sustained Traffic Rate
NLOS	Non-Line-of-Sight
NRM	Network Reference Model
NRTPS	Non Real Time Polling Service
NWG	Network Working Group
OFDMA	Orthogonal Frequency Division Multiplexing Access
PHY	Physical layer
PDU	Protocol Data Unit
PMP	Point-to-MultiPoint
PS	Polling Service
PUSC	Partial Usage of Subcarriers
QoS	Quality of Service
RS	Relay Station

RTPS	Real Time Polling Service
SC	Single Carrier
SDU	Service Data Unit
SFID	Service Flow Identifier
SS	Subscriber Station
SU	Scheduling Unit
TDMA	Time Division Multiplexing Access
TDD	Time Division Duplexing
UL	Uplink
UGS	Unsolicited Grant Service
WiMAX	Worldwide Interoperability for Microwave Access
WMAN	Wireless Metropolitan Area Network

Bibliography

[1] IEEE 802.16 Working Group, "IEEE Standard for Local and Metropolitan Area Networks. Part 16: Air Interface for Broadband Wireless Access Systems," IEEE Standard 802.16-2009, May 2009.

[2] L. Wang, F. Liu, Y. Ji, and N. Ruangchaijatupon, "Admission Control for Non-preprovisioned Service Flow in Wireless Metropolitan Area Networks," in *Proceedings of the Fourth European Conference on Universal Multiservice Networks (ECUMN)*, 2007.

[3] D. Hong and S. S. Rappaport, "Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized handoff procedures," IEEE Trans. Veh. Technol.

[4] H. Wang, W. Li, and D.P. Agrawal, "Dynamic Admission Control and QoS for 802.16 Wireless MAN," in *Proceeding of Wireless Telecommunications Symposium (WTS)*, Pomona, USA, April 2005.

[5] S. Chandra and A. Sahoo, "An Efficient Call Admission Control for IEEE 802.16 Networks," in *Proceeding of IEEE Workshop on Local & Metropolitan Area Networks (LANMAN)*, Princeton, USA, June 2007.

[6] A. Teh and P. Pudney, "Efficient Admission Control Based on Predicted Traffic Characteristics," in *Proceeding of Personal Indoor Mobile Radio Communications (PIMRC)*, Athens, Greece, September 2007.

[7] A. Teh, A. Jayasuriya, and P. Pudney, "Admission Control in Wireless Infrastructure Networks Based on the Predicted Percentage of Delayed Packets," in *Proceeding of Asia-Pacific Conference on Communications (APCC)*, Tokyo, Japan, October 2008.

[8] S. Ghazal, Y.H. Aoul, J.B. Othman, and F. Nait-Abdesselam, "Applying a Self-Configuring Admission Control Algorithm in a New QoS Architecture for IEEE 802.16 Networks," in *Proceedings of IEEE Symposium on Computers and Communications (ISCC)*, Marrakech, Morocco, July 2008.

[9] O.Yang and J.Lu, "Call Admission Control and Scheduling Schemes with QoS support for Real-time Video Applications in IEEE 802.16 Networks," In Journal of Multimedia, May 2006.

[10] "OPNET Simulator," http://www.opnet.com.

[11] F.H.P. Fitzek and M.Reisslein, "MPEG-4 and H.263 Video Traces for Network Performance Evaluation," IEEE Network,Vol. 15,No. 6,pages 40-54, November/December 2001.

[12] "Website Optimization," http://www.websiteoptimization.com/speed/tweak/.

[13] D. C. Schultz P. Herhold H. Yanikomeroglu S. Mukherjee H. Viswanathan M. Lott W. Zirwas M. Dohler H. Aghvami D. D. Falconer R. Pabst, B. H. Walke and G. P. Fettweis, "Relay-based deployment concepts for wireless and mobile broadband radio," IEEE Communications Magazine, vol. 42, pp. 80-89, September 2004.

[14] IEEE 802.16 Working Group, "IEEE Standard for Local and Metropolitan Area Networks. Part 16: Air Interface for Broadband Wireless Access Systems Amendment 1: Multihop Relay Specification," IEEE Standard 802.16j-2009, May 2009.

[15] Y. YU V. Genc, S. Murphy and J. Murphy, "IEEE 802.16j Relay-Based Wireless Access Networks: an Overview," IEEE Wireless Communications, October 2008.

[16] Z. Tao M. Hart and Y. Zhou, "Multihop Cellular Network - Technology and Standardization," International Conference on Computer Communications and Networks (ICCCN), August 2008.

[17] "EU FP7 Project Carrier Grade Mesh Networks (CARMEN), N° 214994," http://www.ict-carmen.eu.