MASTER

A multi-standard simulation platform for hybrid fiber/coax networks

Kwaaitaal, J.J.B.

Award date:
1999

Eindhoven University of Technology

Department of
Electrical Engineering

Division of Telecommunication

Section electro-optical communication systems

A Multi-Standard Simulation Platform
for Hybrid Fiber/Coax Networks

J.J.B. Kwaaitaal

Report of a graduate project carried out
in the period
April 1998 - December 1998
at
Philips Research Laboratory Eindhoven

Supervisors at Philips
ir. S.P.P. Pronk and dr.ir. M.J.M. de Jong
Supervisor at University
ir. H.P.A. van den Boom
Graduate professor
prof.ir. G.D. Khoe

Preface

This report contains the results of the work I have done during my graduation period at Philips Research. I have contributed to a project called "Multi-Standard Simulation Platform for Hybrid Fiber/Coax Networks". This project is carried out in the cluster Access Networks, part of the group Broadband Communication & Video Systems.

The graduation period concludes my study electrical engineering at the Eindhoven University of Technology. As a graduate student, I was a member of the section electro-optical communication systems of prof. ir. G.D. Khoe.

This report can be of particular value to people who are interested in a platform for development of advanced scheduling and medium access control algorithms for hybrid fiber/coax networks. The platform is designed to support different standards. A quick introduction to the way in which advanced scheduling and medium access control algorithms can be implemented is given in paragraph 4.2.1.

For people who are interested in a comparison of the MCNS vs. DVB/DAVIC standards, chapter 5 can be useful. In this chapter, a number of DVB and DAVIC simulations are described, their results are analyzed and compared with an available MCNS simulation of Quigley and Hartman [8].

I would like to thank ir. S.P.P. Pronk and dr.ir. M.J.M. de Jong in particular for their guidance and support at Philips Research. I am also very grateful to ir. H.P.A. van den Boom and prof.ir. G.D. Khoe for supporting me at the University. Furthermore, I want to thank all other people that were willing to help and support me during my graduation project.

Jacco Kwaaitaal, January 1999.

Abstract

The liberalization of the telecommunications market leads to an increase in competition. CATV (community antenna television) network operators seek for new advanced services to offer to the customer. It is foreseen that various services will be provided over hybrid fiber/coax (HFC) networks, such as telephony, internet access, (near) video on demand, interactive services. These services require upgrades of the network to enable bi-directional communication. They have different traffic characteristics and demand different quality-of-service (QoS) levels. To guarantee specific QoS levels, advanced scheduling and medium access control (MAC) algorithms must be developed. Standards for communication in HFC networks are becoming available at the moment. A better understanding of the performance differences between standards is needed. For these purposes, the Multi-Standard Simulation Platform for Hybrid Fiber/Coax Networks (MSSP) is developed.

We concentrate on the following standards: Digital Video Broadcasting[1] (DVB), Digital Audio Video Council 1.3 (DAVIC), IEEE 802.14 (IEEE) and Multimedia Cable Network Standard[2] (MCNS). These standards specify the physical layer and the MAC layer of an HFC network to standardize communication between head-end (HE) and network terminations (NTs), leaving a certain amount of freedom in implementation. We are mainly interested in upstream transmission (NT to HE), where the following mechanisms for medium access are available: (1) ALOHA access, (2) contention tree access, (3) reservation access and (4) fixed access. DVB and DAVIC allow transmission of data and requests in ALOHA. DAVIC and IEEE allow transmission of requests in contention tree. The reservation access is granted as a result of the requests (*request-grant* mechanism). Fixed access is based on periodic grants.

The MSSP is designed hierarchically, following a top-down approach. It consists of a number of levels, which are described in a modular fashion. For flexibility and cost reduction in possible products, based on this system, we have designed a system with low complexity slave NTs that communicate with an intelligent HE. The MAC intelligence and the scheduling algorithms are therefore implemented in the HE. Addition of advanced scheduling algorithms in the HE should not impose changes upon the NTs. In order to simulate different standards within MSSP, we implemented standard-specifics by a number of parameters that can be changed for each simulation. The way in which NTs choose for a particular transmission method is based on the queue status at the NTs and a priority scheme. The scheduling of the upstream channel is divided in a bandwidth allocation part and a grant generation part. The latter determines the specific use of each time slot on the upstream channel. Advanced scheduling strategies can use information on connections and their QoS demands: (1) agreed at connection setup and (2) gathered by monitoring the active connections. Statistics on the contention processes can serve as input to schedulers to optimize allocation of bandwidth to different types of access. From the implementation process of the first scheduling strategies, we conclude that the simulation platform is a flexible tool to develop strategies for advanced scheduling and MAC.

After designing the simulation platform, it was implemented in the simulation environment BONeS (a Cadence product). In this way, we were able to carry out simulations that compare the MCNS vs. DVB/DAVIC standards. We conclude that MCNS has two advantages over the DVB/DAVIC: (1) it makes better use of direct access (data in ALOHA) and (2) it has less transmission overhead costs. The results of this comparison plead for the extension of the simulation platform to support MCNS as well.

---

[1] Recently, DVB took over the specification of DAVIC. For ease of reference, we will continue to use DVB to refer to the old DVB specification.

[2] Not implemented in MSSP yet.

Contents

1 Introduction

The liberalization of the telecommunications market leads to an increase in competition. CATV (community antenna television) network operators seek for new advanced services to offer to the customer. It is foreseen that various services will be provided over hybrid fiber/coax (HFC) networks, such as telephony, internet access, (near) video on demand and interactive services. These services require upgrades of the network to enable bi-directional communication. They have different traffic characteristics and demand different quality-of-service (QoS) levels. To guarantee specific QoS levels, advanced scheduling and medium access control (MAC) algorithms must be developed. Standards for communication in HFC networks are becoming available at the moment. A better understanding of the differences in performance between standards is needed. For these purposes, the Multi-Standard Simulation Platform for Hybrid Fiber/Coax Networks is developed.

The aim of this report is to present the design of this simulation platform and its first simulation results. Although the report gives a complete description of the simulation platform, it is not intended as a specification. Instead, it aims to provide a good understanding of the operation of the simulation platform and the principles it is based on. After designing the simulation platform, it was implemented in the simulation environment BONeS (a Cadence product). In this way, we were able to carry out simulations to compare the MCNS vs. DVB/DAVIC standards.

The report is divided into a number of chapters. A description and model of an HFC network is given in chapter 2. The possibilities of MAC are investigated in chapter 3, along with the degrees of freedom that different standards offer with respect to MAC. Chapter 4 presents the design of the simulation platform that is based on the findings in chapters 2 and 3. The comparison between the MCNS and DVB/DAVIC standards can be found in chapter 5. Finally, some conclusions are drawn in chapter 6.

2 Description and model of a hybrid fiber/coax (HFC) network

2.1 Physical network description

A CATV network is a broadband network, characterized by a tree topology. This tree topology is traditionally used to distribute analog TV signals from the head-end (HE) to the customer premises, called downstream (DS) direction, as illustrated schematically in the black part of Figure 2-1. The depicted network consists of an optical part, from the HE to the district center, and a coaxial part, from the district center to the customer premises. It is therefore called a hybrid fiber/coax (HFC) network. Along the DS path, the signal encounters a number of nodes where it splits into different branches of the network. To increase the signal power received at the customer premises, amplifiers are installed in the network.

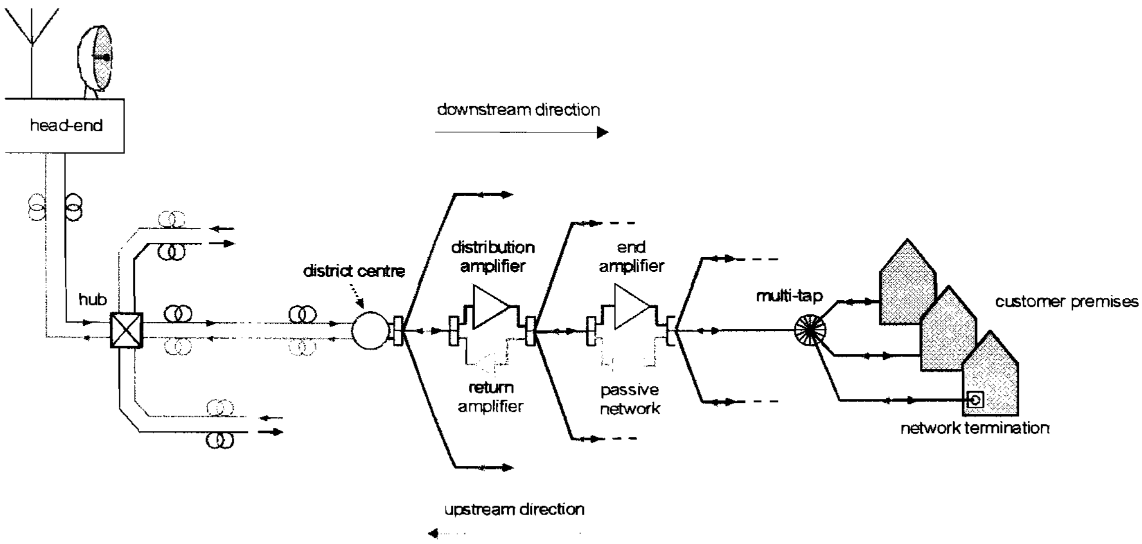


Figure 2-1: Example of an HFC network.

In order to support bi-directional digital communication between the HE and the customer premises, using the existing HFC network infrastructure, some upgrades of the HFC network are required (as depicted in the red part of Figure 2-1). A return path from the customer premises to the head-end, called upstream (US) direction, is created by installing *return amplifiers* in the network. The bandwidth in the US and DS direction is divided into several channels, which cover the frequency range from 5 to 65 Mhz and 110 to 862 Mhz respectively [3]. To achieve this, frequency division multiplexing (FDM) can be used in both DS and US direction. In US direction, code division multiplexing (CDM) may be used as well. Next to the existing analog channels, digital channels will be used to communicate between the HE and *network terminations* (NTs) at the customer premises. These NTs terminate the HFC network and provide the interface between the public access network and the private in-home network.

Before an NT can communicate with the HE, it has to register itself. The NT listens to a DS channel and waits for an opportunity to transmit on a specific US channel, indicated by the HE. Since NTs have different positions in the HFC network, their transmission power has to be adjusted so that the signals from different NTs are received at the HE with equal power. In order to properly synchronize the reception at the HE of the upstream transmission, originating from different NTs, a time offset is applied by the NT to compensate for the delay differences. This offset is based on the reception of DS data. The process of determining this power adjustment and time offset for a specific NT is called ranging. An important limitation, due to the physical implementation of the HFC network, is that the NTs are not able to communicate with each other directly but only through the HE.

2.2 An HFC network model

If we assume that the registration process, including ranging, is carried out successfully, we can define a model of the HFC network as shown in Figure 2-2.

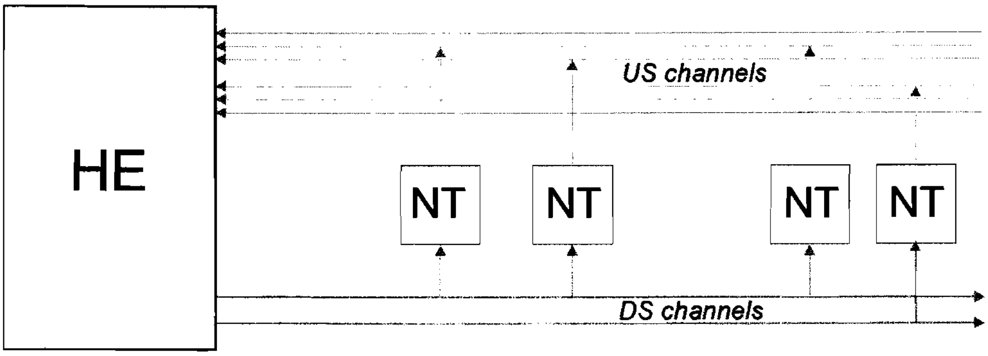


Figure 2-2: Model of an HFC network.

The HFC network model consists of a number of US and DS channels that are managed by the HE. One DS channel may contain management information on several US channels. Grooming, the process of moving communication between HE and NT to different channels, allows flexible usage (e.g. load balancing) of the available channels. For the moment, however, we only consider one DS and one US channel that is shared by a number of NTs. Moreover, we concentrate on the usage of the US channel. This means that we only consider the DS channel for as far as its influence on the US channel is concerned.

Since a US channel is shared by a number of NTs, their US transmissions may collide in which case data will be lost. We will refer to this as a *collision*. Except for collisions in the US direction, we assume that all US and DS transmissions are received error-free by respectively the HE and the NT for which they were destined. A medium access control (MAC) protocol is applied to control the access to a US channel. The US channel time is divided into frames that are subdivided in one or more time slots. A frame denotes a span of time for which the content is explicitly scheduled by the HE. A frame can be of a fixed or variable length. We assume that a complete description of the usage of each slot in a US frame is available at the NT before the start of the frame.

2.3 Standardization

At the moment, several standards become available for HFC networks. We concentrate on the following standards: Digital Video Broadcasting[1] (DVB) [1], Digital Audio Video Council 1.3 (DAVIC) [2], IEEE 802.14 (IEEE) [3] and Multimedia Cable Network Standard[2] (MCNS) [4]. These standards specify the physical layer and the MAC layer of an HFC network to standardize communication between HE and NT, leaving a certain amount of freedom in implementation.

[1] Recently, DVB took over the specification of DAVIC. For ease of reference, we will continue to use DVB to refer to the old DVB specification.

[2] Not implemented in MSSP yet.

DVB, DAVIC and IEEE are based on asynchronous transfer mode (ATM), while MCNS is specifically designed for internet protocol (IP) traffic. We are interested in a comparison of the standards, in terms of throughput, delay and jitter for different types of traffic, and their ability to support certain quality-of-service (QoS) levels. We have to explore the parameter space, accessible within the standards, to find out whether and by what means the standards can be used to guarantee certain QoS levels to the network subscribers. Regarding the simulation platform, we intend to investigate and implement the DVB, DAVIC and IEEE standards first. At a later stage, the platform can be extended to MCNS as well.

3 Medium access control (MAC)

3.1 *Introduction*

Since a US channel is shared by a number of NTs, a MAC protocol is applied to control their access to the US channel. Two basic modes of access are *collisionless* access and *contention-*based access. With collisionless access, only one NT at a time is permitted to transmit in a collisionless slot, allocated by the HE. The HE may allocate these slots periodically for *fixed-*rate data transmission. For variable rate data transmission, these slots are allocated by giving *reservation* grants. These grants result from reservation requests by the NT. It is therefore called the request-grant mechanism. With contention-based access, multiple NTs are permitted to transmit in a contention slot and collisions may occur. If a collision occurs, a retransmission of the data by the NT at a later moment is required. This process is called collision resolution. Contention slots have the advantage of allowing quick access to the channel, but on the other hand yield a low channel utilization. Therefore, they are typically used for transmission of requests and small amounts of data. For an overview of currently available MAC protocols, see [5] and [6].

3.2 *Standards*

In DVB, DAVIC and IEEE, the US channel time is divided into time slots of two different types: ATMslots and minislots. These slots may be either empty of they may contain a protocol data unit (PDU). An ATMslot can contain an ATM PDU (APDU) that typically carries an ATM cell as payload. A minislot can contain a mini PDU (mPDU). In DVB and DAVIC, the US channel consists of a stream of ATMslots, in which a single ATMslot can be subdivided into 3 minislots. In IEEE, the US channel consists of a sea-of-minislots, in which a combination of 4 minislots[1] can form an ATMslot.

Table 1: Frame properties in DVB & DAVIC

physical layer specification	frame length (ms)	ATMslots per frame
US grade A	6	3
US grade B	3	9
US grade C	3	18

DVB and DAVIC have an explicit US frame structure. Some US frame properties are shown in Table 1. IEEE has no US frame structure. However, using variable length frames in IEEE does not impose restrictions upon the US channel. Individual slots in a frame can be allocated for *ALOHA*, *tree*, *reservation* or *fixed* access, if allowed in the standard. In DVB and DAVIC, slots,

[1] Expected value. The value depends on the physical layer overhead and should be an integer number [1].

using the same access mode, should be allocated consecutively so that access mode areas are formed. In addition, these areas should appear in the US frame in the order mentioned above. In IEEE, no such restrictions apply. Reservation and fixed access slots enable collisionless transmissions as described in paragraph 3.1. ALOHA and (contention) tree access enable contention-based access and are described in paragraph 3.3 and 3.4, respectively. A PDU can be used for transmission of a request (R), data (D) or data with a piggybacked request (D+R). *Piggybacking* a request means sending a request along with data in a US slot. Collisionless transmission of a request in a reservation or fixed access slot is called *cycle stealing*. Table 2 describes for each access mode whether or not it is allowed in a specific standard. The table also describes the type of PDU each access mode may contain and how this PDU may be used.

Table 2: An overview of the possible use of specific access modes.

ACCESS MODE	DVB	DAVIC	IEEE	PDU TYPE	USAGE		
					R	D	D+R
ALOHA	■	■		APDU	■	■	
tree		■	■	MPDU	■		
reservation	■	■		APDU	■[a]	■	
			■	APDU	■[a]	■	■[b]
fixed	■	■		APDU	■[a]	■	
			■	APDU	■[a]	■	■[a,b]
		■		MPDU	■[a]	■	

[a] Not excluded in the standard.
[b] Only allowed after reception of a full grant.

3.3 ALOHA protocol

DVB and DAVIC allow the use of ALOHA access. The ALOHA protocol enables direct access to the channel and is defined in the standards as described below. When an NT wants to transmit an APDU, it randomly chooses an ATMslot in the ALOHA area (if present) of the first available US frame. After the APDU is transmitted, the NT waits for feedback from the HE. The feedback indicates whether the HE received the APDU successfully or in collision. If the reception was successful, the ALOHA process is completed and further transmissions can take place.

In case a collision occurred, an exponential backoff procedure is started for controlling a possible retransmission. The NT randomly chooses a number, say n, between 1 and 2^b and retransmits the APDU in the nth ALOHA slot that becomes available, starting at the first ALOHA slot of the next US frame. The parameter b, called the backoff exponent, is equal to **min backoff exponent** for the first retransmission. For further retransmissions, b is increased by 1 after every retransmission. This is done until the backoff exponent reaches the maximum value, **max backoff exponent**. In this case b is not increased further. The procedure is repeated until the APDU is received successfully or the ALOHA process is canceled. The parameters **min backoff exponent** and **max backoff exponent** are set by the HE.

3.4 Contention-tree algorithm

DAVIC and IEEE allow the use of contention tree access. Contention tree access, as defined in in the standards, provides a way to transmit reservation requests in contention. In comparison with ALOHA access, it uses a more structured and flexible approach in resolving collisions, which is explained below. When an NT wants to transmit a request, it randomly chooses a minislot to transmit in. This minislot is chosen from a *group* of m consecutive minislots that represent the root of the contention tree. After transmission, the NT waits for feedback from the

HE. Ternary feedback (empty, success or collision) on the content of each minislot in the group is sent to the NT. If a collision occurred in a minislot, the NTs that collided in this minislot are assigned a new group for retransmission. This group is explicitly mapped on the US channel. The tree continues to expand until all collisions have been resolved. By means of a so-called *mapping method*, the HE controls the order in which groups of a tree are mapped on the US channel. Figure 3-1 illustrates the expansion of a tree and the use of two possible mapping methods. These methods are called *depth first* and *breadth first* mapping. In addition, the standards allow execution of several trees in parallel.

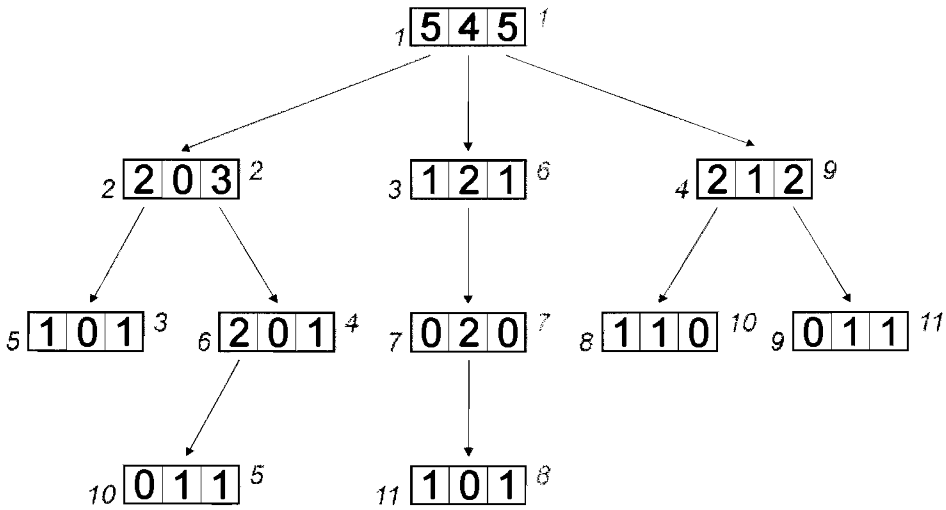


Figure 3-1: Example of contention tree expansion with m=3. The *depth first* mapping is indicated in red (right), the *breadth first* mapping in blue (left). The number of contenders in a minislot are indicated in black.

There are two important differences between the DAVIC and IEEE standards: (1) the number of minislots that may be assigned per group, called the **splitting factor**, (2) the way in which newcomers may gain access to a contention tree. In DAVIC, the **splitting factor** is fixed at 3, unlike in IEEE where it is dynamically chosen from 1 to 256 for the root group and from 1 to 16 for subsequent groups. If access to a contention tree is only allowed in the root, then the contention tree is said to be **blocked**. This is the case in IEEE. When newcomers may join the contention tree in any group, the tree is called **non-blocked**. In DAVIC and IEEE, both **blocked** and **non-blocked** trees are allowed. In IEEE, access to the root group is restricted by means of **priority classes** and an **admission time boundary**. Priorities enable discrimination between high- and low-priority NTs. A priority mask indicates which out of 8 priority classes are allowed to enter the root. The **admission time boundary** specifies before which time a request must have become pending (i.e. waiting to be transmitted) in order to enter the root. DAVIC follows a different approach to restrict access to the root group by using a parameter, called **entry spreading**. At the start of a tree, the NT chooses a number, say k, between 1 and **entry spreading** (3..16384). If $k \leq 3$, the kth minislot in the root group is used for transmission. Otherwise, the NT should wait for the start of a new tree.

3.5 Sustaining connections

We can distinguish two types of connections: (1) fixed rate and (2) variable rate. After the connection setup, the HE periodically allocates collisionless slots for a fixed rate connection without the need for NTs to send reservation requests. In DVB and DAVIC, a single grant for the periodical use of a specific slot in a US frame is given to an NT. In IEEE, a fixed rate connection is sustained by periodically giving grants for the use of a slot to an NT. The latter allows more flexibility in the position and rate of granted slots in a US frame.

Variable rate connections are generally sustained by using the request-grant mechanism. At startup, a request is transmitted in contention via ALOHA or tree access. There are two ways to achieve low delays for requests in contention: (1) minimizing collisions by assigning sufficient bandwidth for contention and (2) reducing the number of request messages by using them efficiently. As a result of reservation requests, the NT receives grants for US slots. A request can be granted in several separate grants (partial grants) or completely (full-grant). A full-grant also refers to the last partial grant needed to fully grant a request. Further requests can be transmitted collisionless, if allowed. This can be done by means of piggybacking or cycle stealing and thus yielding a low request delay. Piggybacking in IEEE is only allowed after a full-grant. For cycle stealing to be efficient, it should be used moderately, reducing the number of wasted slots. To avoid the use of contention access and allowing support for certain QoS levels, the HE can send unrequested grants to an NT. This is possible in DVB and DAVIC but not in IEEE. In addition to the request-grant mechanism, DVB and DAVIC allow transmission of data using ALOHA access. This provides a low access delay, if not too many collisions occur. An NT is allowed to send data in ALOHA, when no more than **maximum contention access length** number of data cells is in its queue. When the number of data cells in its queue exceeds **maximum contention access length**, an NT is obliged to send a request using either ALOHA or tree access.

4 Multi-standard simulation platform for hybrid fiber/coax networks (MSSP)

4.1 Purpose

The main purpose of the MSSP is to analyze the performance of MAC and scheduling algorithms in order to develop more advanced algorithms. These algorithms should be able to efficiently support a number of traffic sources (applications) with varying QoS demands. The MSSP should provide knowledge on the similarities and differences between the operation of the MAC protocol in the standards DVB, DAVIC, IEEE and MCNS. Therefore, these standards must be implemented, including a range of allowed settings that are not fixed herein. Furthermore, the performance changes that result from extensions to the standards (e.g. piggybacking) should be evaluated and used as input for the standardization committees. The simulations are aimed to investigate the traffic generation, the transmission of MAC messages, and the US data flow in the HFC network. The output of the simulations should provide accurate estimations regarding the performance of various MAC and scheduling algorithms.

4.2 Design

4.2.1 General considerations

The MSSP is designed hierarchically, following a top-down approach. It consists of a number of levels, which are described in a modular fashion. Each level in the architecture is described by its modules and the interfaces between them. The modules and `interfaces` are indicated by their respective fonts. It is noted that each module can have a different number of architectural levels depending on its complexity. For the naming of submodules, an acronym of the name of the parent module is used as a prefix (e.g. NT receiver). A complete and detailed description of the timing aspects within the design can be found in [9].

For flexibility and cost reduction in possible products, based on this system, we intended to design a system with low complexity slave NTs that communicate with an intelligent HE. The MAC intelligence and the scheduling algorithms should therefore be implemented in the HE. Adding advanced scheduling algorithms in the HE should not impose changes upon the NTs. We intend to encapsulate standard-specific functionality in only a few modules at the lowest appropriate level. In order to simulate different standards within MSSP, we implement standard-specifics by a number of parameters that can be changed for each simulation.

To fill-in the US frame structure, the HE may use information available about the traffic behavior of applications. This information can be obtained in two ways: (1) from agreements on the traffic behavior during connection setup and (2) from statistics gathered on the actual offered traffic. Furthermore, the HE can gather statistics on the contention processes and dynamically adapt their properties in order to optimize their performance. The scheduling process is divided into a bandwidth allocation part and a grant generation part. A grant generation module is allowed to fill-in part of the US frame for specific use, given by the bandwidth allocation module. The grant generation module informs the bandwidth allocation module about the slots that it has occupied. Further cycles or rounds (bandwidth allocation + grant generation) may be needed to complete the whole US frame. Different grant generation modules are typically used for different access modes. For ALOHA access no distinction between grants exists. However, for tree and reservation access the grant generator should decide which tree groups or connections to serve and in what order, respectively. Furthermore, the grant generator may decide to start new trees and, if allowed, dynamically change their properties (e.g. splitting factor).

In order to keep the NT simple and independent from HE innovations, the NT does not make assumptions about the US frame structure. Furthermore, the NT does not monitor the US channel to gather statistics on: (1) the available bandwidth for different access modes and (2) the performance of the contention channel. The NT solely decides to use a specific transmission method based on the queue status, i.e. the number of data cells in the queue. The *transmission method* is defined as a specific combination of access mode and its usage (e.g. data in ALOHA). A priority scheme prescribes the order in which different transmission methods are preferably used. Note that some transmission methods may not be allowed or possible in a specific standard or under certain conditions. In this case, the next highest transmission method that qualifies is chosen. When there is no need for sending requests, transmission methods with this aim are skipped. More details about conditions that restrict the use of a specific transmission method are given in paragraph 4.2.4. The transmission methods that apply are given below in order of decreasing priority:

1. *data in fixed area*[1]
2. *piggyback in ALOHA*[2]
3. *data in ALOHA*[2]
4. *piggyback in reservation*[3]
5. *cycle stealing*[3]
6. *data in reservation*[3]
7. *request in tree*
8. *request in ALOHA*

[1] Requires the connection to be of fixed rate type.

[2] Since we intend to use every reservation grant given to an NT, direct access is not used when reservation grants are available or upcoming. Otherwise, as long as the contention process is active, grants will have to be discarded to prevent changes in order of arrival, doubling or loss of ATM cells.

[3] Requires the availability of reservation grants for the NT.

4.2.2 Level-1 design

The level-1 architecture of the system is shown in Figure 4-1. The HE, HFC network, and the NTs jointly comprise the access network. For the moment, only one application per NT is considered.

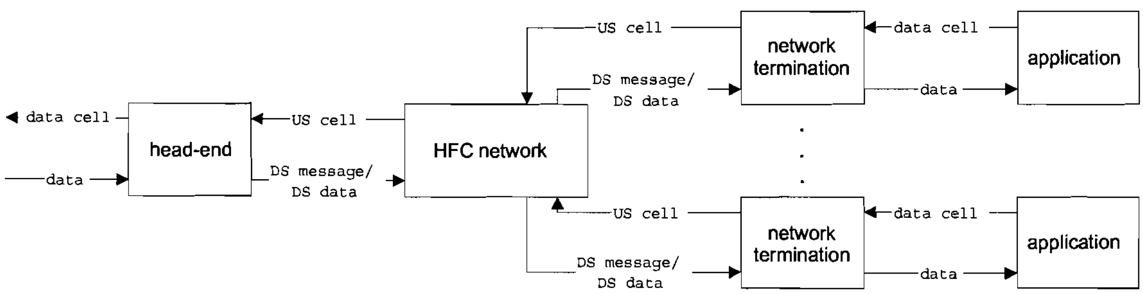


Figure 4-1: Level-1 architecture of the simulation platform.

The access network provides a transparent interconnect between the applications and a core network (attached to the HE). An NT accepts `data cells` from the application. For US transmission of these `data cells`, the NT shares the US channel with all other NTs in a slotted, time-multiplexed fashion.

Access to the US channel is contention-based or collisionless. The NT may, besides transmitting `data cells` as `US cells`, generate `requests`. If allowed by the standard, these `requests` may be sent as `US cells` or as a piggyback on `US cells` that contain `data cells`. For convenience, an NT transmits an empty `US cell` if no other `US cell` is to be transmitted.

In US direction, the HFC network multiplexes `US cells`. If more than one non-empty `US cells` are transmitted simultaneously, the HFC network produces a `collision`. In DS direction, the HFC network broadcasts `DS messages` and `DS data` to all NTs.

The HE accepts the `US cells` or `collisions` and outputs the `data cells` to the core network. The `requests` and `collisions` are processed. Furthermore, the HE controls the usage of the US channel. The HE transmits `DS messages`, which may contain `contention feedback`, `grants` and `frame info`. The latter is used mainly for timing and framing purposes. The `DS messages` are time-multiplexed with `DS data`. The `DS data` contains `data` (from the core network) and has been added for reasons of completeness. For the time being, they do not constitute any functionality in the simulation model.

4.2.3 Level-2 design

4.2.3.1 NT architecture

The level-2 architecture of the network termination is given in Figure 4-2.

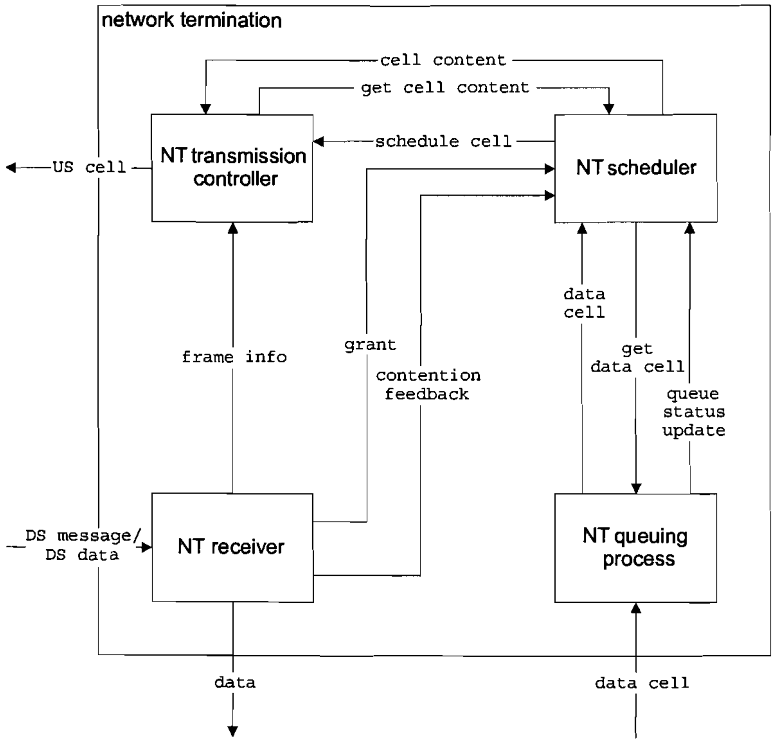


Figure 4-2: Level-2 architecture of the network termination.

The NT receiver receives `DS data` and `DS messages`. The `DS data` is filtered and sent to the application as `data`. The `DS messages`, that contain information about the usage of the US channel, are passed through to the NT transmission controller and the NT scheduler.

The `data cells` from the application are accepted and queued by the NT queuing process in a single first-in-first-out (FIFO) queue. The number of newly enqueued `data cells` is reported to the NT scheduler.

The NT scheduler is the only module in the NT that contains standards-specifics and has two important functions: (1) it schedules the transmission of `US cells`, based on the queue status, the `contention feedback` and the `grants`, (2) it delivers the content of the `US cell` upon a `get cell content` input. The `cell content` can be a `request`, a `data cell` or both, in case of piggybacking. When needed, a `data cell` is obtained from the NT queuing process by means of a `get data cell`.

The NT transmission controller takes care of the actual transmission of the `US cells` that were scheduled. It is the only module that contains time-dependent functionality as opposed to the other modules, which are fully input-event driven. Separation of scheduling and actual transmission allows the decision on the actual content of the `US cell` to be postponed until just before transmission.

4.2.3.2 HFC network architecture

Figure 4-3 shows the level-2 architecture of the HFC network.

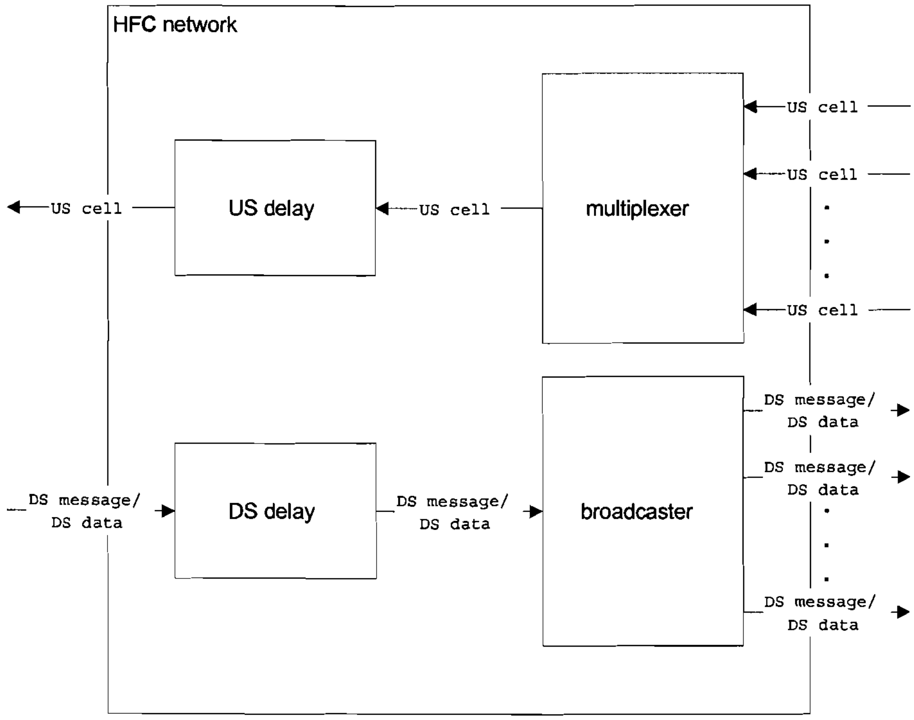


Figure 4-3: Level-2 architecture of the HFC network.

The HFC network models the actual medium. It has a multiplexing functionality in the US direction and a broadcast functionality in the DS direction. Furthermore, it introduces some delay. Multiplexing is done by using the cardinality associated with each `US cell`. An empty `US cell` has cardinality 0, a single `US cell` has cardinality 1 and a `collision` has cardinality 2 or more. In the latter case, the cardinality represents the number of `US cells` in collision. If the sum of the cardinalities of all `US cells` to be multiplexed in the same timeslot is 0, an empty `US cell` is forwarded. If it is 1, then the single `US cell` is forwarded. Otherwise, a `collision` is forwarded. This method allows NTs to send `collisions` to the

HFC network. It enables a future extension to 'super NTs', that represent more NTs in one module. Broadcasting is done by copying the `DS messages` and `DS data` to the different outputs.

The US and DS delays depend on the size of the message, the bandwidth of the channel and the propagation delay. The delay δ_M that a message M of size l incurs on a channel with transmission rate B is given by

$$\delta_M = \tau_{prop} + \tau_M = \tau_{prop} + \frac{l}{B}, \tag{4.1}$$

where τ_{prop} denotes the propagation delay (≈5 µs/km) and τ_M the message delay.

4.2.3.3 HE architecture

Figure 4-4 gives the level-2 architecture of the head-end.

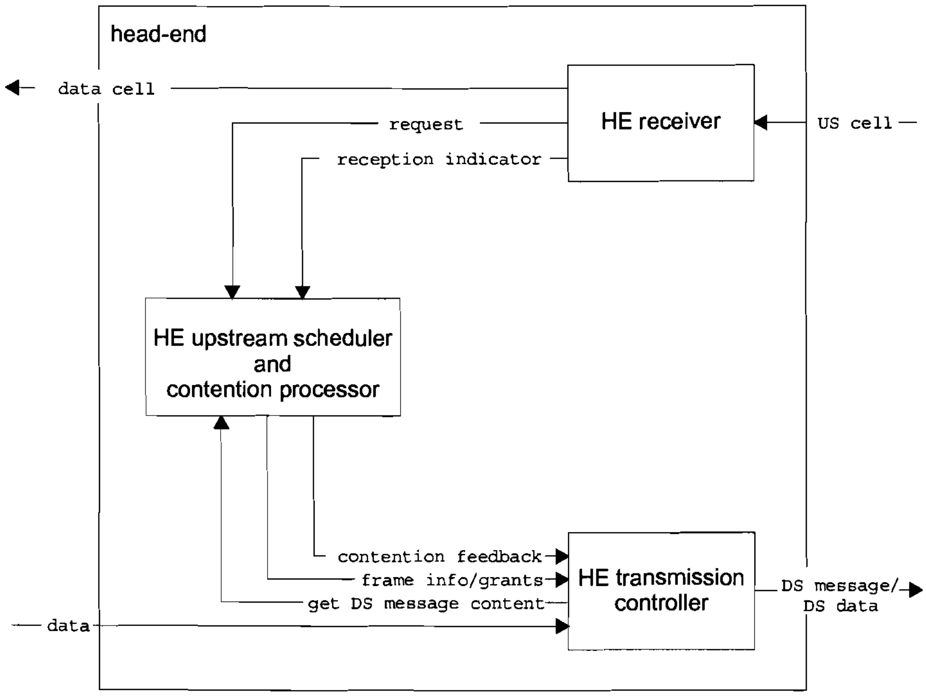


Figure 4-4: Level-2 architecture of the head-end.

The HE receiver accepts `US cells` coming from the HFC network. When a `US cell` arrives successfully and contains a `data cell`, this `data cell` is forwarded to the core network. If it contains `requests`, then these are sent to the HE upstream scheduler and contention processor just like the `reception indicator`. The latter contains information about the upstream slot status (empty, success or collision). Actually, the `reception indicator` contains the whole `US cell` that can be used for statistics and monitoring purposes.

The HE upstream scheduler and contention processor used to consist of two modules with some interfaces in between. Later, it turned out that large portions of information had to be available in both modules. Therefore, it was better to integrate them and use a shared memory interface. As a consequence, the functionality of the HE upstream scheduler and contention processor comprises: (1) scheduling of the upstream bandwidth, (2) controlling the contention processes and (3) generating the `frame info/grants` and `contention feedback`.

The **HE transmission controller** controls the actual transmission of a `DS message`. This message contains the `frame info`, `grants` and `contention feedback` for a particular frame. The **HE transmission controller** is the only module that contains time-dependent functionality, while all the other modules in the HE are fully input-event driven.

4.2.4 Detailed design of the NT

In this paragraph, we present the **NT scheduler** in more detail. The tasks of the **NT scheduler** comprise the scheduling of cells and, just before transmission, the delivery of their content. In Figure 4-5, the level-3 architecture of the **NT scheduler** is shown.

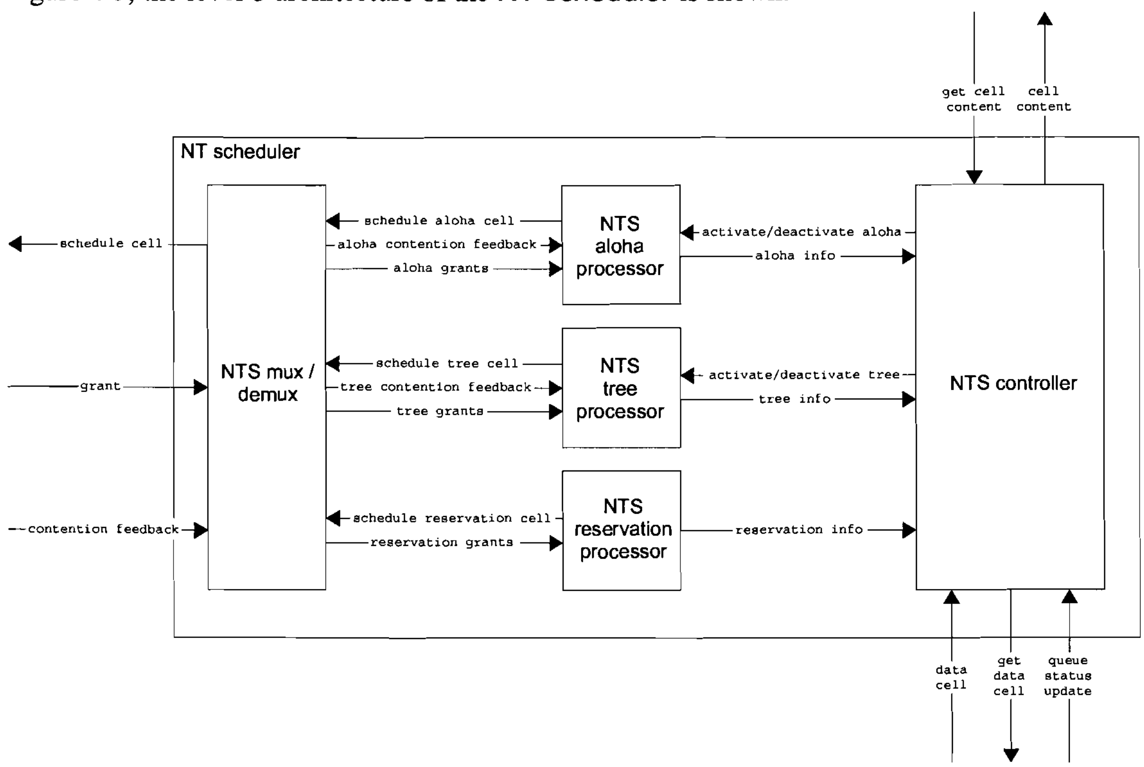


Figure 4-5: Level-3 architecture of the NT scheduler.

The **NTS mux/demux** demultiplexes the `grant` and `contention feedback`, received from the HE, to the appropriate processor. Each access mode is handled differently, which results in an **NTS aloha processor, NTS tree processor** and **NTS reservation processor.** Notice that in MSSP, we implement fixed rate connections by periodically giving `reservation grants`. This does not impose restrictions upon the actual implementation within a particular standard. The **NTS reservation processor** filters the `reservation grants` meant for the specific NT and schedules all granted cells for transmission. `Reservation info`, which contains the number of scheduled cells, is sent to the **NTS controller**. As opposed to the **NTS reservation processor**, both **NTS aloha processor** and **NTS tree processor** need to be activated by the **NTS controller** in order to start processing `aloha grants/aloha contention feedback` or `tree grants/tree contention feedback`, respectively. In this way, scheduling of slots can be disabled in case no ALOHA or tree slots are needed or allowed and also computational cost is reduced. The **NTS aloha processor** takes care of the ALOHA transmission process. It hides the specific actions from the NTS controller (e.g. choosing an ALOHA slot, exponential backoff) that are needed for sending data or requests in ALOHA. The NTS aloha processor gives `aloha info` to the NTS controller with simple values like success, collision and cell scheduled. This indicates the progress in the ALOHA transmission process. The NTS aloha processor can be deactivated externally by the NTS controller or becomes inactive (idle) automatically after a successful transmission. A similar approach applies for the NTS tree processor. State diagrams, that describe the operation of these processors in detail, can be found in Appendix A.

The NTS controller contains: (1) the standard-specifics, implemented by means of parameters and (2) the intelligence to control the transmission of `data cells`. To determine an appropriate transmission method, the NTS controller uses the priority scheme mentioned in paragraph 4.2.1. The status of different processes is tracked by means of status variables. The **status variables** and *rules* that apply (indicated by their respective fonts) are described in Table 3 and Table 4, respectively.

Table 3: NT status variables

PURPOSE	STATUS VARIABLE	VALUE
tree status	**tree**	active/active locked/inactive
ALOHA status	**aloha**	active/active locked/inactive
	aloha transmit mode	data/request/piggyback
reservation status	**cells scheduled**	number of reservation cells scheduled
fixed rate status	**fixed rate**	active/inactive
data transmission status	**transmission**	blocked/non-blocked
	data cells contending	number of data cells in contention
queue status	**queue length**	number of data cells in the queue
request status	**requests contending**	number of requests in contention
	requests pending	number of requests pending

The **status variables** and values in Table 3 that do not clarify themselves are explained below. The **tree** and **aloha** variable represent the status of the NTS tree processor and the NTS aloha processor, respectively. The value "active locked" indicates that restrictions apply in deactivating the processor. It is assigned when a contention cell has been scheduled. When the contention cell is not actually sent yet, the processor can be deactivated. If the contention cell has already been sent, the NTS controller is not allowed to deactivate the processor. The latter restriction is needed, because it would otherwise remain unclear whether or not the cell is received successfully by the HE. When **fixed rate** is "active", the NT will not send requests to the HE. It expects the HE to give grants at a specific rate. The **transmission** is set to "blocked" when data is sent in ALOHA and no contention feedback is received yet. In this state, it is not allowed to send data in reservation to prevent changes in order of arrival, doubling or loss of ATMcells. **Data cells/requests contending** contains the number of data cells/requests that are underway in contention with no contention feedback received yet. **Requests pending** contains the number of successfully requested reservation cells (i.e. requests that are successfully received at the HE) that are not granted yet.

Table 4: NT parameters

ACCESS MODE	PARAMETERS \ STANDARDS	*DVB*	*DAVIC*	*IEEE*
ALOHA	*aloha allowed*	Yes	Yes	No
	max contention access length	variable	variable	-
	max request size for aloha	255	255	-
tree	*tree allowed*	No	Yes	Yes
	max request size for tree	-	255	4095
ALOHA/tree	*boundary for additional requests*	variable	variable	variable
reservation	*piggybacking allowed*	No	No	Yes
	max request size for PB	-	-	15
	PB allowed on aloha PDUs	-	-	-
	PB request boundary	-	-	0
	cycle stealing allowed	Yes[a]	Yes[a]	No
	max request size for CS	255	255	-
	CS request boundary	<15	<15	-

[a] Not excluded in the standard.

The **status variables** of Table 3 and the *parameters* in Table 4 can be used to build conditions that restrict the use of specific transmission methods. Most parameters in Table 4 should be self-explaining. A few will be explained below. The *aloha allowed* parameter, for example, can completely switch off the use of ALOHA access. In this case, transmission method 2,3 and 8 in the priority scheme from paragraph 4.2.1 are disabled. When ALOHA is allowed, however, the queue length may exceed the *max contention access length*. Then, data in ALOHA is still not allowed, but requests in ALOHA are permitted as explained in paragraph 3.5. The *boundary for additional requests* is used in the following condition:

$$\frac{\textbf{queue length - cells scheduled}}{\textbf{requests pending}} > \textit{boundary for additional requests}. \qquad (4.2)$$

Additional requests in contention are only permitted if condition (4.2), representing a minimum relative queue growth, is met. This reduces the burden on the contention channel. Similarly, the *CS request boundary* is introduced in the following condition:

$$\textbf{requests pending} + \textbf{requests contending} \leq \textit{CS request boundary} \ . \qquad (4.3)$$

Because cycle stealing wastes a reservation slot, it is done only when the number of requests, pending at the HE, drops below a certain threshold. For efficiency, cycle stealing should be done at the latest possible moment to gather as much requests as possible in one message. However, to make sure that the data transmission is not held because of request delay, requests should arrive at the HE before **requests pending** becomes zero. The *PB request boundary* is introduced to allow implementation of the IEEE standard. In this standard, piggybacking is only allowed after a full-grant, which we implement with the following condition (with *PB request boundary* equal to zero for IEEE):

$$\textbf{requests pending} + \textbf{requests contending} \leq \textit{PB request boundary} \ . \qquad (4.4)$$

Piggybacking does not waste reservation slots and the restriction above is therefore a bit strange. It would be better to use piggybacking whenever requests need to be done. This results in a more up-to-date status at the HE of the queues at the NTs. Flowcharts, that describe the operation of the NTS controller in detail, can be found in Appendix B.

The transmission process, controlled by the NTS controller, consists of two parts: activation and content delivery. Activation of the NTS aloha processor or NTS tree processor is needed to initiate scheduling of ALOHA and tree cells, respectively. When no contention access is needed, the activation part is skipped. The need for activation of a contention processor is checked after every `queue status update` and after successful transmission in contention. The latter is needed, because a `queue status update` might have occurred during an active contention process, in which case activation is not possible. This can prevent newly arrived `data cells` from being requested. Furthermore, due to the introduction of the *boundary for additional requests*, it can happen that after a successful transmission in contention, the relative growth of the queue is too small to allow immediate re-activation. To make sure, however, that newly arrived `data cells` are always requested in the end, a final check for the need of activation is made after `reservation info` (reception of reservation grants). This is done by applying condition (4.2) again.

Upon a request by the NT transmission controller just before the actual transmission, the NTS controller delivers the `cell content`. It has to decide whether to send a request, data, piggyback or not use the transmit opportunity at all. The decision is based on the values of **status variables** and *parameters* and is according to the priority scheme mentioned in paragraph 4.2.1. A reason for not using a reservation cell can be an empty data queue. Not using a contention cell can be due to a higher priority transmission method that became available (e.g. piggybacking).

4.2.5 Detailed design of the HE

In this paragraph, we take a closer look at the HE upstream scheduler and contention processor (HEUSCP). To recapitulate, the tasks of the HEUSCP comprise the scheduling of the upstream bandwidth as well as controlling the contention processes. In Figure 4-6, the level-3 architecture of the HEUSCP is shown. Below, the functionality of the submodules within the HEUSCP is described.

The memory, shared between the HEUSCP contention processor and the HEUSCP upstream scheduler, can be divided in three categories. The HEUSCP basic memory contains several tables that are needed for basic operation to store the attributes and status of different processes. The HEUSCP statistics memory is used for gathering statistics on the US channel. It can be used by the HEUSCP upstream scheduler to implement advanced scheduling strategies. For example, the ratio between empty, successful and collided cells in the ALOHA area of US frames can be used as input for the HEUSCP upstream scheduler. The scheduler may subsequently increase or decrease the bandwidth allocated for ALOHA access. The HEUSCP monitoring memory is meant for storing information to enable offline analysis of US channel usage. For the moment, only the HEUSCP basic memory is considered.

The HEUSCP contention tree queues module contains one queue for each active contention tree. These queues are filled with `group requests` by the HEUSCP contention processor in order to request the scheduling of new groups for contention tree expansion. The HEUSCP upstream scheduler can issue a `dequeue group request` for a particular contention tree by indicating its queue.

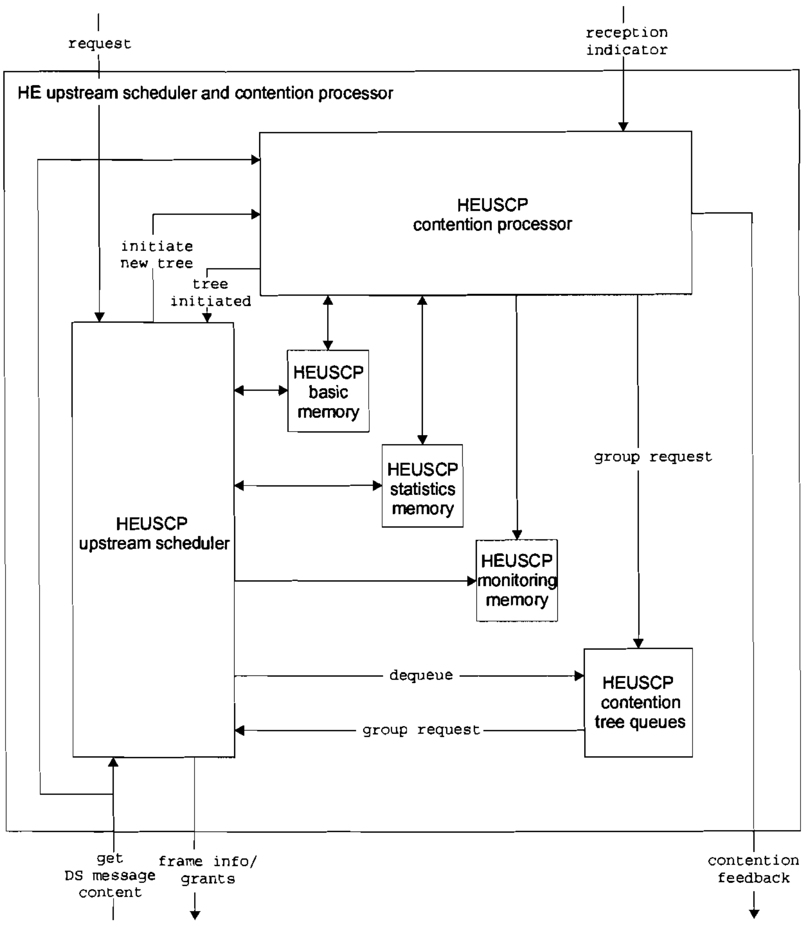


Figure 4-6: Level-3 architecture of the HE upstream scheduler and contention processor.

The HEUSCP upstream scheduler contains the standard-specifics, the attributes and status of connections, and the intelligence of the HE. These aspects are jointly described later on in this paragraph. It accepts the (reservation) `requests` of connections and updates their status accordingly. The scheduling process is initiated by a `get frame info/grants`. It consists of bandwidth allocation and grant generation. A specific grant generator is allowed to fill-in part of the US frame indicated by the bandwidth allocator. To generate `aloha grants`, no additional information or actions are needed as opposed to the generation of `tree grants` and `reservation grants`. To start filling-in the part of the US frame allocated for tree access, the HEUSCP upstream scheduler removes completed trees from the list of active trees. It can also decide to `initiate new trees` while determining their attributes (e.g. blocked/non-blocked). This initiates the generation of `group requests`. Then, it can `dequeue` a number of `group requests` from active trees and allocate a number of minislots per group (i.e. splitting factor) to generate `tree grants`. The selection of the trees to serve, can be based on the attributes (e.g. priority) or status (e.g. number of levels) of the trees. Filling-in the reservation area of an US frame can be based on the attributes (e.g. fixed rate or QoS requirements) and status (e.g. number of requests pending) of connections. As a result of the scheduling process, `frame info/grants` are sent back to the HE transmission controller and the tables in the HEUSCP basic memory are updated to match the new state in the system.

The HEUSCP contention processor controls the contention process. Upon arrival, a US cell is identified by its first minislot number that makes it possible to derive all the information needed for action from the tables in the HEUSCP basic memory. For ALOHA slots, a feedback message is created that contains the slot status of each ALOHA slot in a particular US frame. This message is sent to the HE transmission controller per US frame as required in the standards. For tree slots no such restriction applies. A feedback message for these slots is created containing the slot status of the tree slots that arrived between each `get DS message content`. In the latter case, feedback information may be gathered until just before DS transmission. When a collision occurred in a tree slot, the feedback message also contains an identifier for a new group in which the NTs may retransmit. A request for the scheduling of this group is put in the corresponding contention tree queue, immediately after receiving the collision. This implicitly determines a breath first approach for tree resolution, because new groups are requested in the same order as collisions in tree slots appear on the US channel. Completion of a tree as well as other status updates are stored in the HEUSCP basic memory. An `initiate new tree` triggers the start of a new tree. Subsequently, a `group request` for the root group is placed in the assigned contention tree queue. Then, the tree initialization process is concluded with a `tree initiated`.

Figure 4-7 shows the level-4 architecture of the HEUSCP upstream scheduler. The submodules should be self-explaining after reading the functionality description above. Below, the scheduling process of the US channel is described in more detail.

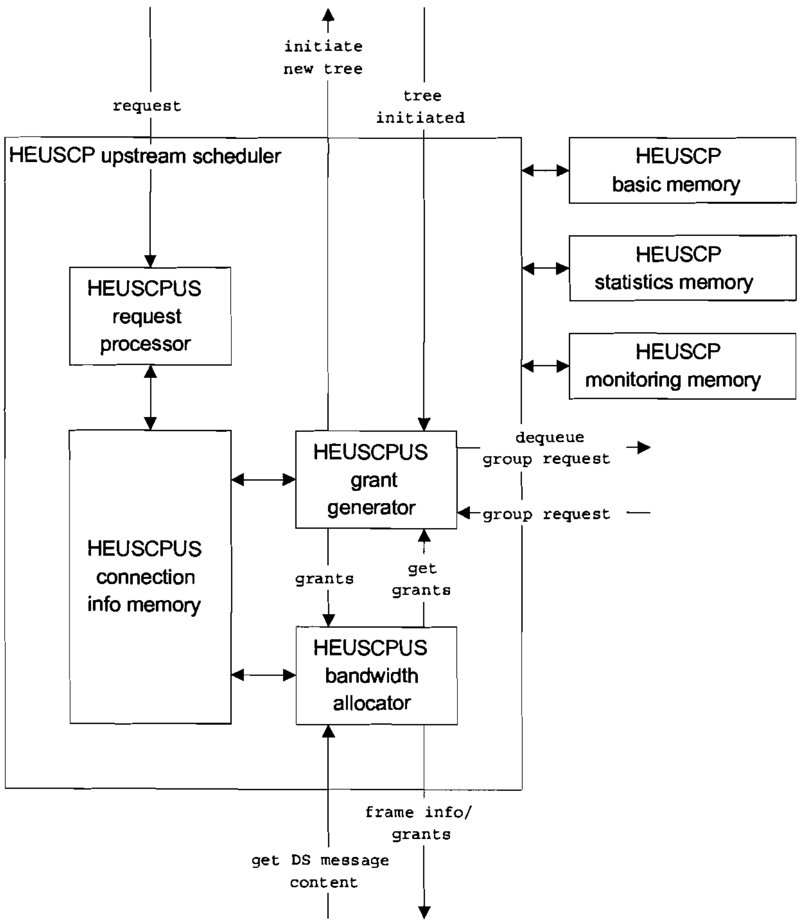


Figure 4-7: Level-4 architecture of the HEUSCP upstream scheduler.

The intelligence of the HE to control the scheduling process is located in the HEUSCP upstream scheduler. To be able to implement advanced scheduling techniques, we designed an upstream scheduler that easily allows upgrades in bandwidth allocation and grant generators. Furthermore, a bandwidth allocation strategy, accompanied by grant generators that fill-in the slot usage, can be chosen by simply setting some simulation parameters. This allows a quick comparison of different scheduling strategies for a particular set of connections.

If it is triggered by a `get DS message content`, the **HEUSCPUS bandwidth allocator** sends a `get grants` to the **HEUSCPUS grant generator**. This `get grants` contains: (1) a bitmap of the US frame that indicates which of the slots are used/unused, (2) the particular grant generator (e.g. reservation grant generator) that is allowed to fill-in unused slots and (3) the maximum number of slots the grant generator is allowed to fill-in. The **HEUSCPUS grant generator** can determine the best position of specific grants in the US frame. This can be used to guarantee a minimum cell delay variation within a specific connection, for example. A so-called *partial schedule* is concluded with sending `grants` back to the **HEUSCPUS bandwidth allocator**. More cycles or rounds may follow invoking various grant generators.

The bandwidth allocation strategies (BASs) together with the corresponding grant generators that have so far been implemented in MSSP, are shown in Figure 4-8.

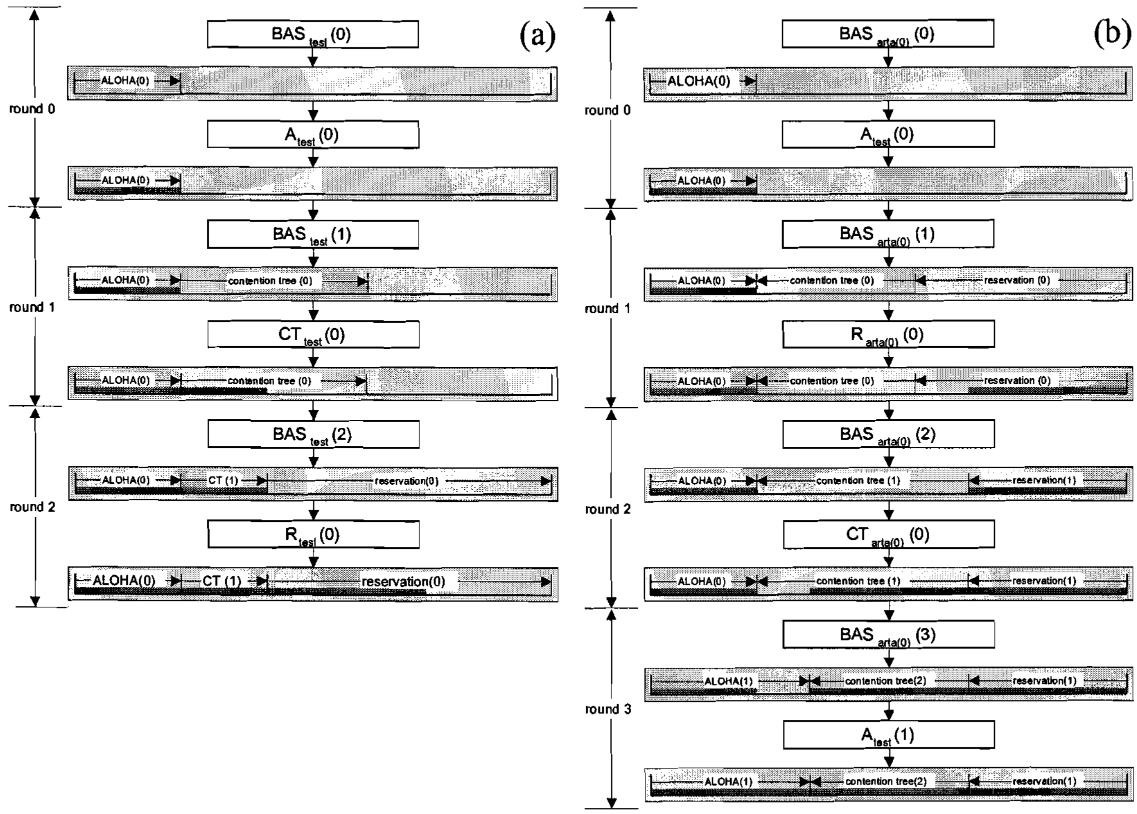


Figure 4-8: Illustration of the bandwidth allocation strategies: (a) *TEST* and (b) *ARTA*(0). The ALOHA, tree and reservation grant generators are indicated by A( ), CT( ) and R( ), respectively. The number between brackets identifies a particular invocation.

The first BAS, called *TEST*, allocates a fixed size area within the US frame for ALOHA, tree and reservation. In this order the grant generators are successively invoked to fill-in the US frame as illustrated. The second BAS, called *ARTA*(0), allocates a minimum area for ALOHA and contention tree. The remainder of the US frame may be used for reservation access. When the reservation grant generator can not fill-in all slots it is allowed to use, the contention tree generator is allowed to use them. Otherwise, they will be used to expand the ALOHA area size. The reservation grant generator and the tree grant generator serve the connections and trees in a round-robin fashion, respectively. For connections this is done per ATMcell and for trees this is done per group. Furthermore, at most one new tree is initiated per US frame.

Standards may impose some restrictions on the bandwidth allocation and grant generation. The standard-specifics are implemented by means of parameters, which can be found in Table 5.

Table 5: HE parameters

ACCESS MODE	PARAMETERS \ STANDARDS	*DVB*	*DAVIC*	*IEEE*
general	*frame size fixed*	Yes	Yes	No
	frame size (minislots)	9/18/54	9/18/54	-
	respect ATMslot boundaries	Yes	Yes	No
	number of minislots per ATMslot	3	3	4[a]
ALOHA	*aloha allowed*	Yes	Yes	No
	aloha at start of frame	Yes	Yes	-
tree	*tree allowed*	No	Yes	Yes
	blocked tree access fixed	-	No	Yes
	blocked tree access	-	-	Yes
	entry spreading allowed	-	Yes	No
	splitting factor fixed	-	Yes	No
	splitting factor	-	3	-
reservation	*unrequested grants allowed*	Yes	Yes	No

[a] Expected value. The value depends on the physical layer overhead and should be an integer number, see [1].

Most parameters are self-explaining, however, a few remarks are made below. The *frame size fixed* parameter is used in combination with the *frame size* parameter. When the former is true, the latter determines its value. Otherwise, the latter parameter will not be used. Some other parameters, like *splitting factor*, use the same principle. The *respect ATMslot boundaries* parameter only allows an ATMslot frame structure in which, per ATMslot, a conversion can be made to minislots. This excludes, for example, the appearance of one minislot between two ATMslots in DVB and DAVIC.

Besides allocating slots and determining their position (possibly with some restrictions), the contention trees and connections to serve have to be chosen. This choice is based on their status and priority. At the moment, the HEUSCPUS connection info memory only contains the status of the number of requests pending per connection. To take advantage of the differences between connection requirements, each connection should be assigned a number of attributes. The scheduler can use these attributes to serve each connection tailored to its needs. To take advantage of additional information for schedulers, a more thorough investigation on advanced scheduling is needed, which is outside the scope of this work.

4.2.6 Applications

An application generates packets. Since a packet may exceed the size of an ATMcell payload (48 bytes), it is divided into a number of segments that each fit in an ATMcell payload before they are delivered to the access network. In the applications described here, the segmentation is done according to the ATM adaptation layer specification 5 (AAL5), which adds an 8-byte trailer to the packet. Currently, all segments are placed in the NT queue in zero time. However, the application is optionally extended with a traffic shaper. This shaper can use a peak cell rate algorithm, for example [7]. In the current version of the simulation platform an NT is connected to a single application.

Two applications that are implemented at the moment are a *Poisson* packet generation process and an *IEEE 802.14* packet generation process. The IEEE 802.14 packet generation process aims to model ethernet traffic [10]. Both Poisson and IEEE 802.14 applications generate packets according to a Poisson process. These packets have negative exponentially distributed inter-arrival times. The mean packet generation rate is given by λ. This parameter is used to vary the load that an application offers. Differences between both types of applications are: (1) the size of the packets generated and (2) their probability of occurrence. Both values are given in Table 6 along with the number of resulting segments or ATMcells.

Table 6: Packet size distribution with resulting number of ATMcells for the Poisson and IEEE 802.14 application.

APPLICATION	PACKET SIZE (BYTES)	PROBABILITY	NUMBER OF ATMCELLS
Poisson	40	1	1
IEEE 802.14	64	0.60	2
	128	0.06	3
	256	0.04	6
	512	0.02	11
	1024	0.25	22
	1518	0.03	32

Many more applications are being built that aim to model real-life applications, such as telephony over IP, FTP, WWW, telnet and video applications.

5 Simulations

5.1 Aims

The aim of the first set of simulations was to verify the operation of the platform using the *TEST* strategy/grant generators and Poisson applications. The results of these simulations proved a proper operation of the platform. Details of the verification process are not included in this document.

The second set of simulations, carried out in MSSP, aims to compare the performance of the DVB & DAVIC standards to that of the MCNS standard. The performance is measured in terms of mean packet transmission delay, i.e. the mean time between packet generation and delivery of the packet to the core network. The performance is observed as a function of the aggregate load of the applications.

Thomas J. Quigley and David Hartman of Broadcom Corporation carried out a performance simulation of the MCNS standard [8]. Our simulations of the DVB and DAVIC standards are carried out in MSSP under similar conditions. The results are compared and some suggestions that could explain the differences in performance are made. The simulations do not intend to be exhaustive, but should merely be regarded as a first result of the MSSP. At the same time, they can serve as a starting-point for further deployment of MSSP as a tool for designing advanced scheduling strategies.

5.2 Assumptions

The values of the system parameters for both the MCNS simulation and the DVB/DAVIC simulations are given in Table 7.

Table 7: Values of the system parameters for the MCNS and DVB/DAVIC simulations.

PARAMETER	MCNS	DVB/DAVIC
Upstream bandwidth	1.28 Mbps	1.28 Mbps
Downstream bandwidth	1.28 Mbps	1.28 Mbps
Network length	50 km	50 km
Number of NTs	20	20
Minislot size	16 bytes	21 bytes
HE scheduling discipline	First-in-first-out (packet)	Round-robin per ATMcell
ALOHA backoff exponent	min = 2; max = 16	min = 2; max = 16
Data in ALOHA	Enabled	Enabled
Direct access size	64 bytes packet	1 ATMcell
Interleaving DS message	Disabled	Disabled

In order to allow a fair comparison of our simulations with the one of Quigley and Hartman, we use the same US and DS channel bandwidth for our simulations. These US and DS channel bandwidths are actually not allowed in DVB/DAVIC. To achieve a US frame length of about 3 ms, which is common in DVB/DAVIC, we decided to let each US frame consist of 8 ATMslots (3.2 ms). The bandwidth of the DS channel, carrying the DS messages, is assumed to be large compared to its load. This means that the access delay does not change significantly with changes in DS message load. Furthermore, it is important to notice from Table 7, that for direct access in the MCNS simulation, a packet of 64 bytes can be transmitted in its entirety. In DVB/DAVIC this is done per ATMcell.

Further properties of the system that influence the system performance are: (1) the type of applications connected, (2) the type of scheduling used and (3) the overhead costs. The NTs in the system are connected to IEEE 802.14 applications as described in paragraph 4.2.6. The DVB/DAVIC simulations use the *ARTA*(0) scheduling strategy illustrated in Figure 4-8b. Apart from the scheduling differences in the simulations, like division of contention and reservation bandwidth, the MCNS and DVB/DAVIC standards have different overhead costs (e.g. forward error correction, guard time). Table 8 shows the resulting size of packets due to the overhead costs.

Table 8: Resulting packet sizes due to transmission overhead in MCNS and DVB/DAVIC for the IEEE 802.14 application.

PROBABILITY	NET PACKET SIZE (bytes)	GROSS PACKET SIZE IN MCNS (bytes)	GROSS PACKET SIZE IN DVB/DAVIC (bytes)
0.60	64 (40[a])	112	128 (64[a])
0.06	128	256	192
0.04	256	496	384
0.02	512	752	704
0.25	1024	1232	1408
0.03	1518	1712	2048
AVERAGE	368.1	476.8	531.2
NET US CHANNEL BANDWIDTH	-	988 Kbps	887 (919[a]) Kbps

[a] In some simulations, we used adapted IEEE 802.14 applications, in which the smallest packet size was reduced to 40 bytes to fit in one ATMcell. The values of this adapted application that differ from the original are given between brackets.

5.3 Simulations and results

The simulations carried out in MSSP can be grouped in DVB-simulations (b,c,d) and DAVIC-simulations (e,f,g,h). For comparison convenience, the MCNS simulation of Quigley and Hartman (a) is also included in the figures. All simulations use IEEE 802.14 applications, unless stated differently. Each simulation is briefly described below:

MCNS: (a) MCNS simulation of Quigley and Hartman [8].

DVB: (b) DVB simulation with an ALOHA and reservation area. The *min. aloha area* is set to 2 ATMslots (of 8 ATMslots). Data in ALOHA (direct access) is allowed, but restricted by a *max contention access length* of 2.

(c) Same as (b), but data in ALOHA is not allowed.

(d) DVB simulation with an ALOHA and reservation area. The *min. aloha area* is set to 1 ATMslot (of 8 ATMslots). Data in aloha is not allowed and adapted IEEE 802.14 applications are used.

DAVIC: (e) DAVIC simulation with a tree and reservation area. The *min. tree area* is set to 2 ATMslots (of 8 ATMslots). Only 1 blocked tree is available for contention access.

(f) Same as (e), but with 4 blocked trees available for contention access.

(g) Same as (f), but with minimum tree area set to 1 ATMslot (of 8 ATMslots).

(h) Same as (g), but with adapted IEEE 802.14 applications.

Figure 5-1 and Figure 5-2 show the results of the simulations for DVB and DAVIC, respectively.

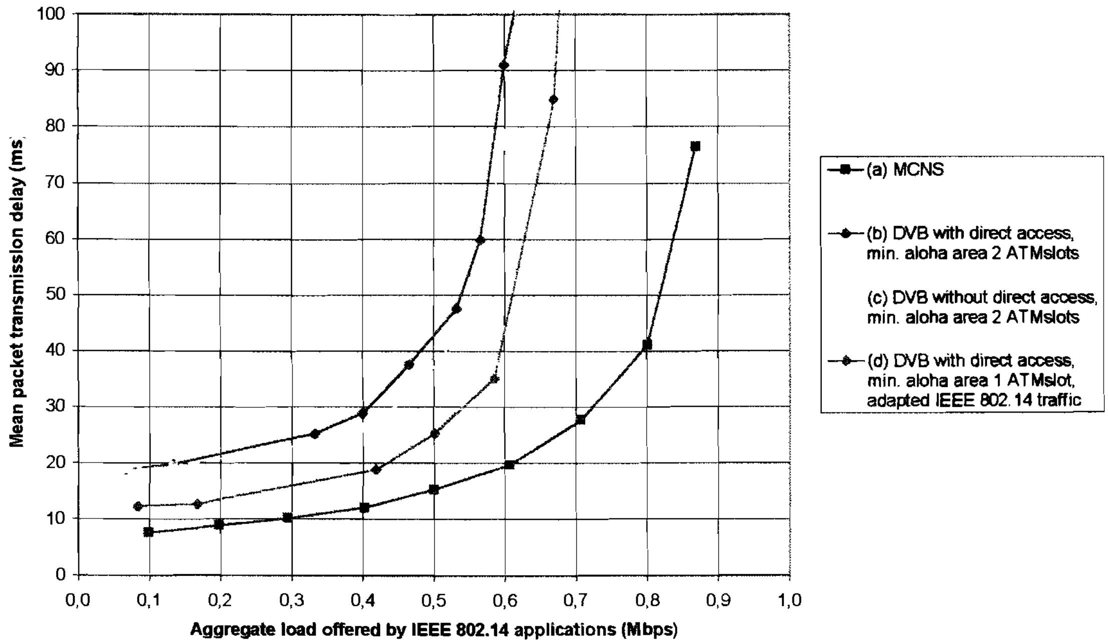


Figure 5-1: Performance comparison of the DVB and MCNS standards.

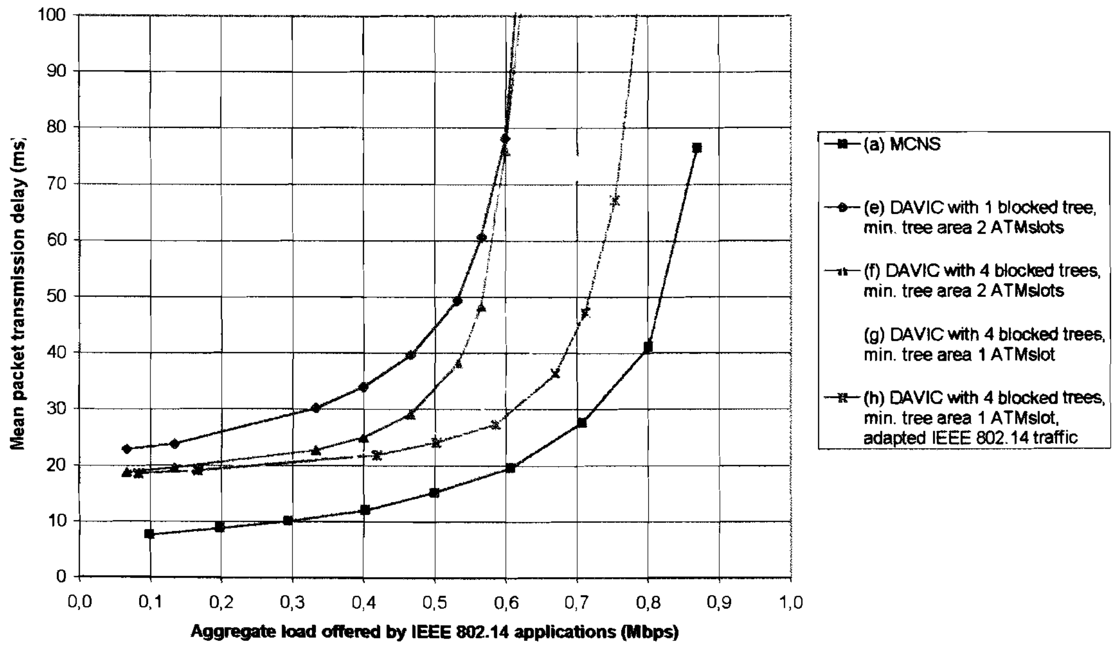


Figure 5-2: Performance comparison of the DAVIC and MCNS standards.

5.4 Analysis

In this paragraph, the results of the DVB and DAVIC simulations are analyzed. First, two important aspects of the results are investigated: (1) the minimum value of the mean packet transmission delay that occurs at low loads and (2) the value of the offered load at which the packet transmission delay increases rapidly. Second, we look at the mutual differences between DVB simulations and those between DAVIC simulations, which can not be explained by the aspects above. Third, some conclusions regarding the performance of the standards are drawn.

The approach that is used to determine the minimum value of the mean packet transmission delay is described below. The DVB simulations from Figure 5-1 are used as an illustration. However, a similar approach is valid for the other simulations. When a packet arrives in the queue at the NT, the NTS aloha processor is activated. It waits for the arrival of a `DS message` that contains `aloha grants`. Since the frame size in our simulations is 3.2 ms, the mean waiting time is given by $\tau_1 = 1.6$ ms. Some NT processing time is needed ($\tau_2 = 0.5$ ms). At low loads, the US frame only consists of ALOHA slots. Since an ALOHA cell is randomly scheduled within the US frame, the mean time between the start of the frame and the cell transmission is given by $\tau_3 = 1.6$ ms. The time needed for US transmission is given by (4.1) and depends on the message size. An APDU (64 bytes) takes $\tau_4 = \tau_{prop}+\tau_M = 0.3+0.4 = 0.7$ ms to transmit. This APDU can contain: (1) a request or (2) the first segment of a packet, if direct access is allowed. We assume that it arrives at the HE between two `DS messages`. Then, the waiting time for the next `get DS message content` is given by $\tau_5 = 1.6$ ms. The `DS message` contains `reservation grants` for sending (more) packet segments in APDUs. Some HE processing time is needed ($\tau_6 = 1.0$ ms). The DS transmission takes $\tau_7 = \tau_{prop}+\tau_M = 0.3+3.2\cdot10\% = 0.6$ ms. Some NT processing time is needed again (τ_2). Because the *ARTA*(0) reservation grant generator starts filling-in reservation cells from the end of the US frame, this would add an extra time of approximately 3.2ms for small packets (60%). This results in a mean delay per packet of approximately $\tau_8 = 3.2\cdot60\% = 2$ ms. Assume that the whole packet can be transmitted in consecutive US frames. Then, the US transmission takes $\tau_9 = \tau_{prop}+\tau_M$, where τ_M is determined using a message size equal to the packet size (incl. overhead). The minimum value of the mean packet transmission delay can now be calculated by summing all delays mentioned and taking the probability of particular packet sizes into account. The resulting minimum values of the mean packet transmission delay for IEEE 802.14 applications and adapted IEEE 802.14 applications are 17 ms and 12 ms, respectively. These values correspond well with the minimum values of the mean packet transmission delay in Figure 5-1. When cells are scheduled near the beginning of a US frame this gives an improvement of about 3.5 ms for normal IEEE 802.14 applications and 2 ms for the adapted ones. Notice that the better performance of the adapted IEEE 802.14 application is mainly caused by the use of direct access for the smallest packet. This packet can be transmitted completely within one ALOHA slot.

The throughput of the system determines the value of the offered load at which the packet transmission delay increases rapidly. To show this, an upper bound for the throughput of the system in the DVB and DAVIC simulations is determined below. Two aspects that decrease the useful US channel bandwidth of the system are: (1) the transmission overhead mentioned in Table 8 and (2) the bandwidth needed for transmission of requests. From Table 8, we see a decrease of the net US channel bandwidth to 887 Kbps for normal and 919 Kbps for adapted IEEE 802.14 applications. This is the bandwidth that remains for requests and data. Under high loads, we can assume that contention slots are only used for requests (the *max contention access* *length* is likely to be exceeded in this situation). Therefore, the bandwidth available for data transmission is further reduced by the amount of bandwidth assigned to contention access. Under high loads, the contention area is reduced to the *min. aloha area* or *min. tree area* for DVB and DAVIC, respectively. This results in the following upper bounds of the throughput:

$\frac{6}{8} \cdot 887\,\text{Kbps} = 665\,\text{Kbps}$ for simulation (b), (c), (e) and (f),

$\frac{7}{8} \cdot 887\,\text{Kbps} = 776\,\text{Kbps}$ for simulation (g),

$\frac{7}{8} \cdot 919\,\text{Kbps} = 804\,\text{Kbps}$ for simulation (d) and (h).

In Figure 5-1 and Figure 5-2, we see that the mean packet transmission delay increases rapidly for loads near the upper bound of the throughput as expected.

The difference between DVB simulation (b) and (c) can be understood by taking a look at the transmission of the smallest packet (segmented in 2 ATMcells). In simulation (b), this packet can be transmitted using direct access. However, an NT must wait for feedback, indicating a successful transmission of the first segment, before transmission of the second segment is permitted. The resulting delay at low loads is comparable to the delay that the request-grant mechanism takes in simulation (c). At medium loads, however, more collisions occur due to the increased load on the contention channel because of the data transmissions. Therefore, direct access even decreases the performance as shown in Figure 5-1.

In DAVIC simulation (e) with one blocked tree, the transmission of requests can be blocked for longer periods. Furthermore, tree slots are sometimes wasted while waiting for feedback. Using more trees in parallel, simulation (f), gives more opportunities to enter the tree contention process. Furthermore, one tree can resolve, while another is waiting for feedback and thus wasting less tree slots. This results in lower access delays.

We conclude from Figure 5-1 and Figure 5-2 that the MCNS standard gives the best performance under the current circumstances. Two explanations for this are: (1) a smaller amount of transmission overhead and (2) a better use of direct access compared to DVB and DAVIC. Scheduling cells near the beginning of a US frame can improve the performance of the DVB and DAVIC implementations. Optimizing the minimum contention area can further improve their performance. Options like cycle stealing and piggybacking can be enabled to gain even more, because they relieve the burden on contention slots.

6 Conclusion

In this report that presents the design of the Multi-Standard Simulation Platform for Hybrid Fiber/Coax Networks, special attention was given to the way in which it supports different standards. Furthermore, the way in which the simulation platform is able to implement advanced scheduling and medium access control (MAC) algorithms is described. Advanced scheduling strategies can use information on connections and their quality-of-service (QoS) demands, agreed at connection setup. They can also use statistical information gathered by monitoring the active connections. Statistics on the contention processes can serve as input to schedulers to optimize allocation of bandwidth to different types of access. From the implementation process of the first scheduling strategies, we conclude that the simulation platform is a flexible tool to develop strategies for advanced scheduling and MAC.

From the analysis of the simulations that aim to compare MCNS vs. DVB/DAVIC standards, we conclude that MCNS has two advantages over the DVB/DAVIC: (1) it makes better use of direct access and (2) it has less transmission overhead costs. Improvements in the performance of the DVB/DAVIC simulations can be achieved by: (1) optimizing the position of transmissions within the upstream frame, (2) optimizing the bandwidth allocated to contention access, (3) using options like cycle stealing and piggybacking to relieve the burden on the contention slots. The results of this comparison plead for the extension of the simulation platform to support MCNS as well.

Abbreviations and acronyms

AAL5	ATM adaptation layer specification 5 (for packet segmentation)
APDU	ATM protocol data unit
ARTA(0)	bandwidth allocation strategy (aloha-reservation-tree-aloha, version 0)
ATM	asynchronous transfer mode
BAS	bandwidth allocation strategy
CDM	code division multiplexing
DAVIC	Digital Audio/Visual Council 1.3 standard
DS	downstream
DVB	Digital Video Broadcasting standard
FDM	frequency division multiplexing
FIFO	first-in-first-out
HE	head-end
HFC	hybrid fiber/coax
IEEE	IEEE 802.14 standard
IP	internet protocol
MAC	medium access control
MCNS	Multimedia Cable Network Standard
mPDU	mini protocol data unit
MSSP	Multi-Standard Simulation Platform for Hybrid Fiber/Coax Networks
NT	network termination
PDU	protocol data unit
QoS	quality-of-service
US	upstream

References

[1] European Telecommunications Standards Institute (ETSI)[1]
DIGITAL VIDEO BROADCASTING; DVB INTERACTION CHANNEL FOR CABLE TV DISTRIBUTION SYSTEMS (ETS 300 800)
France, Valbonne, Sophia Antipolis, April 1998.
[2] Digital Audio-Visual Council[1]
DAVIC 1.3 SPECIFICATION PART 8; LOWER LAYER PROTOCOLS AND PHYSICAL INTERFACES
Switzerland, Geneva, March 1998.
[3] IEEE Standards Committee[1]
IEEE PROJECT 802.14/A DRAFT 3 REVISION 3 - CABLE-TV ACCESS METHOD AND PHYSICAL LAYER SPECIFICATION.
USA, New York, Institute of Electrical And Electronics Engineers, October 1998.
[4] Cable Television Laboratories, Inc.[1]
DATA-OVER-CABLE SERVICE INTERFACE SPECIFICATIONS RADIO FREQUENCY INTERFACE SPECIFICATION (MCNS)
Cable Television Laboratories, Inc., July 1998.
[5] Tanenbaum, Andrew S.
COMPUTER NETWORKS
The Netherlands, Amsterdam, Vrije Universiteit
USA, New Jersey, Upper Saddle River, Prentice-Hall, Inc., 1996 third edition.
[6] Bertsekas, D. and R. Gallager
DATA NETWORKS
Massachusetts Institute of Technology
USA, New Jersey, Upper Saddle River, Prentice-Hall, Inc., 1992 second edition.
[7] Stallings, W.
ISDN AND BROADBAND ISDN WITH FRAME RELAY AND ATM
USA, New Jersey, Englewood Cliffs, Prentice-Hall, Inc., 1995 third edition.
[8] Quigley, Thomas J. and David Hartman
PERFORMANCE SIMULATION OF THE MCNS MEDIA ACCESS CONTROL PROTOCOL APPLIED TO THE OPEN CABLE SIGNALING CHANNEL
CA, Irvine, Broadcom Corporation, December 1997.
[9] Pronk, S.P.P and M.J.M. de Jong
MULTI-STANDARD SIMULATION PLATFORM FOR HYBRID FIBER/COAX NETWORKS; I. STANDARDS SURVEY, II. BASIC ARCHITECTURE. (TN179/98)
Netherlands, Eindhoven, Philips Research, July 1998.
[10] IEEE 802.14 Working Group
EVALUATION MODELS FOR IEEE 802.14 MAC PROTOCOL (IEEE 802.14/95-061R2)
IEEE 802.14 Working Group, January 1996.

[1] Note that these documents are drafts or interim specifications that are still under development.

Appendix A: State diagrams of the NT processors

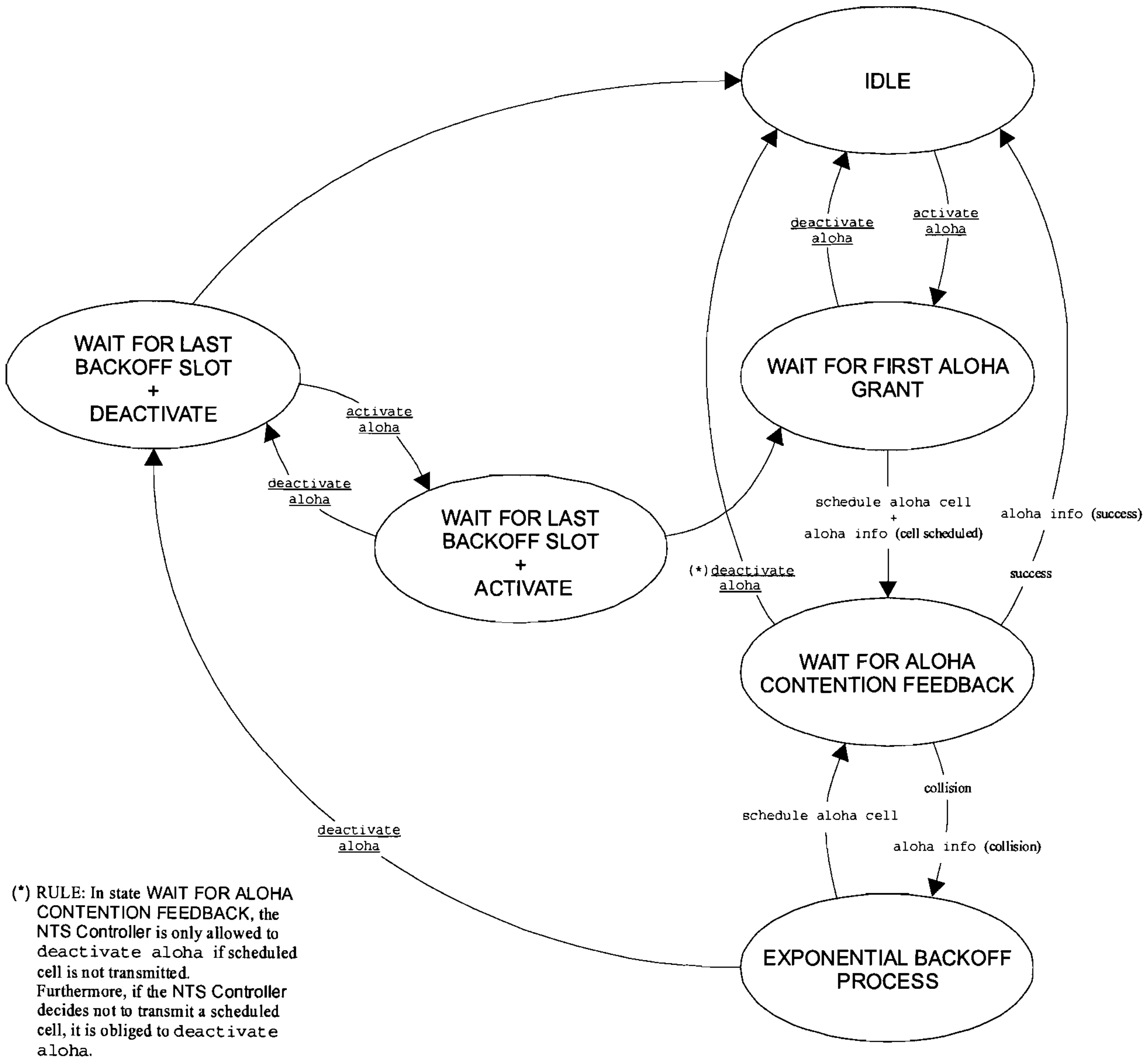
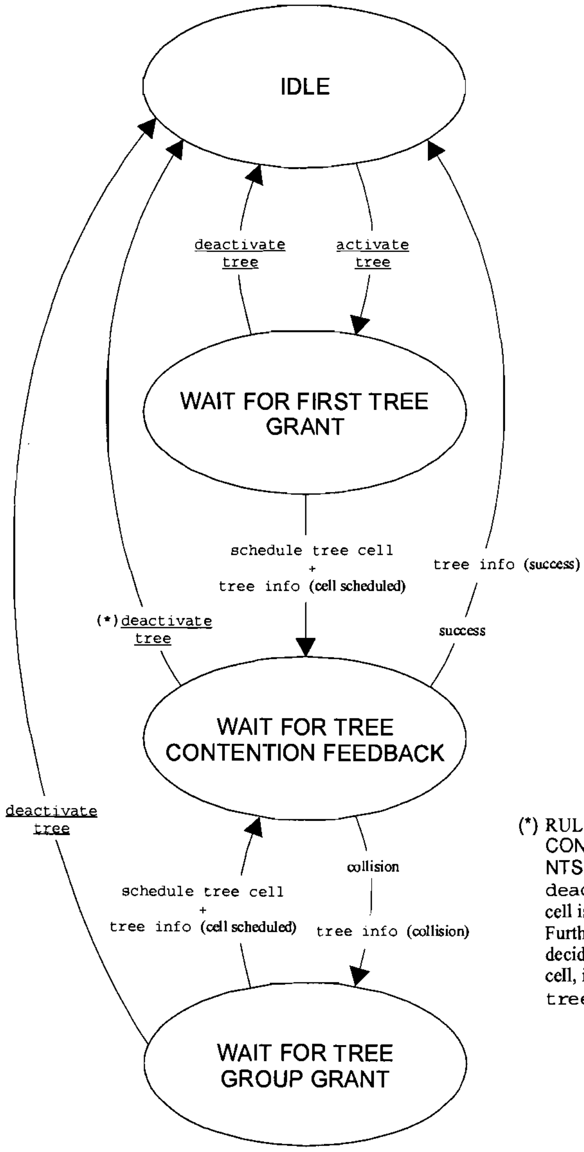


Figure A-1: State diagram of the NTS aloha processor



(*) RULE: In state WAIT FOR TREE CONTENTION FEEDBACK, the NTS Controller is only allowed to `deactivate tree` if scheduled cell is not transmitted. Furthermore, if the NTS Controller decides not to transmit a scheduled cell, it is obliged to `deactivate tree`.

Figure A-2: State diagram of the NTS tree processor

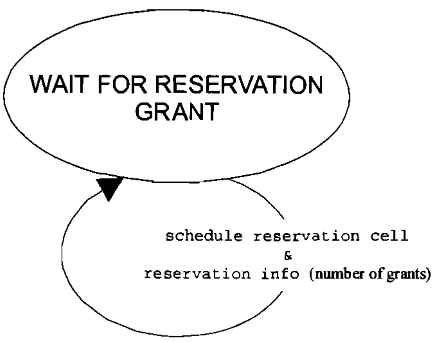


Figure A-3: State diagram of the NTS reservation processor

Appendix B: Flowcharts of the NTS controller operation

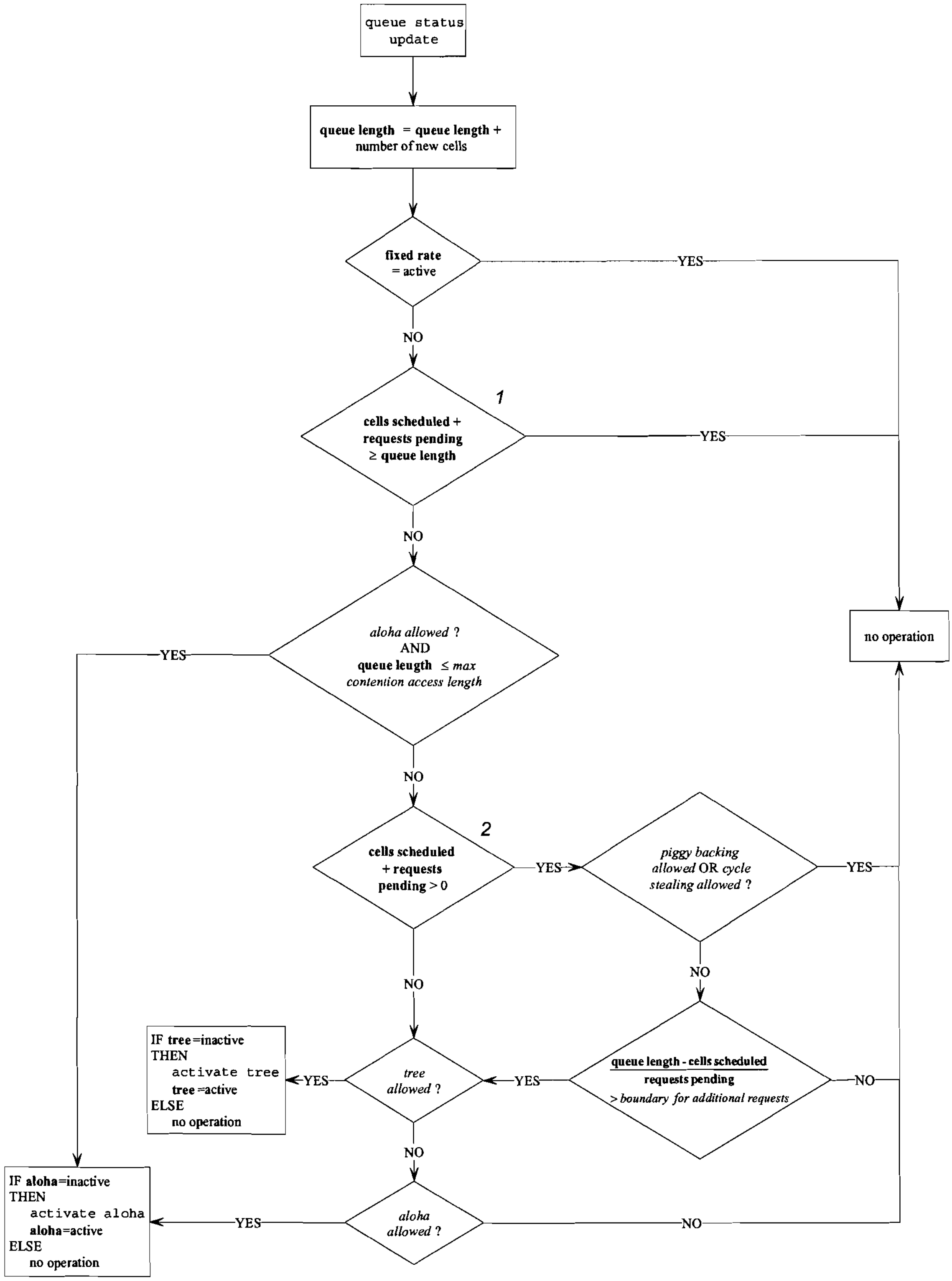


Figure A-4: Flowchart of decisions and actions after a `queue status update`**.**

Notes concerning the flowchart in Figure A-4:

ad. *1*: When there is no need for sending new requests, then do not activate a processor.

ad. *2*: If the condition is satisfied, this means that reservation cells are scheduled or upcoming.

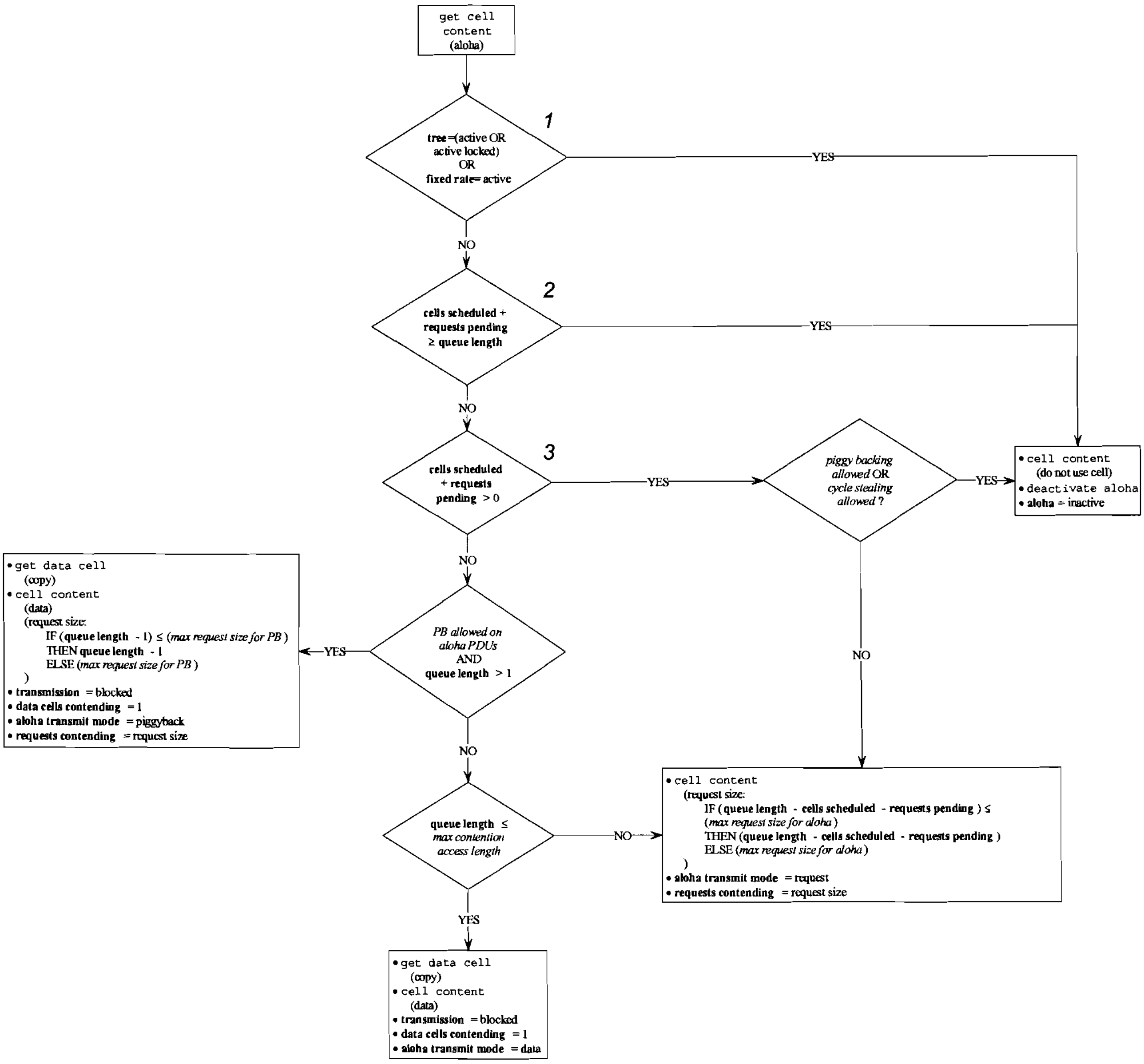


Figure A-5: Flowchart of decisions and actions after a `get cell content` (aloha).

Notes concerning the flowchart in Figure A-5:

ad. *1*: When **tree** is active or active locked, it is preferred to send requests in tree access mode. Furthermore, data is not sent in ALOHA, because reservation slots might remain unused if ALOHA contention feedback takes too long.

ad. *2*: When there is no need for sending new requests, then do not use the ALOHA slot.

ad. *3*: If the condition is satisfied, this means that reservation cells are scheduled or upcoming.

When data is transmitted in ALOHA the status variable **transmission** is set to blocked, to prevent data cells from being sent in reservation.

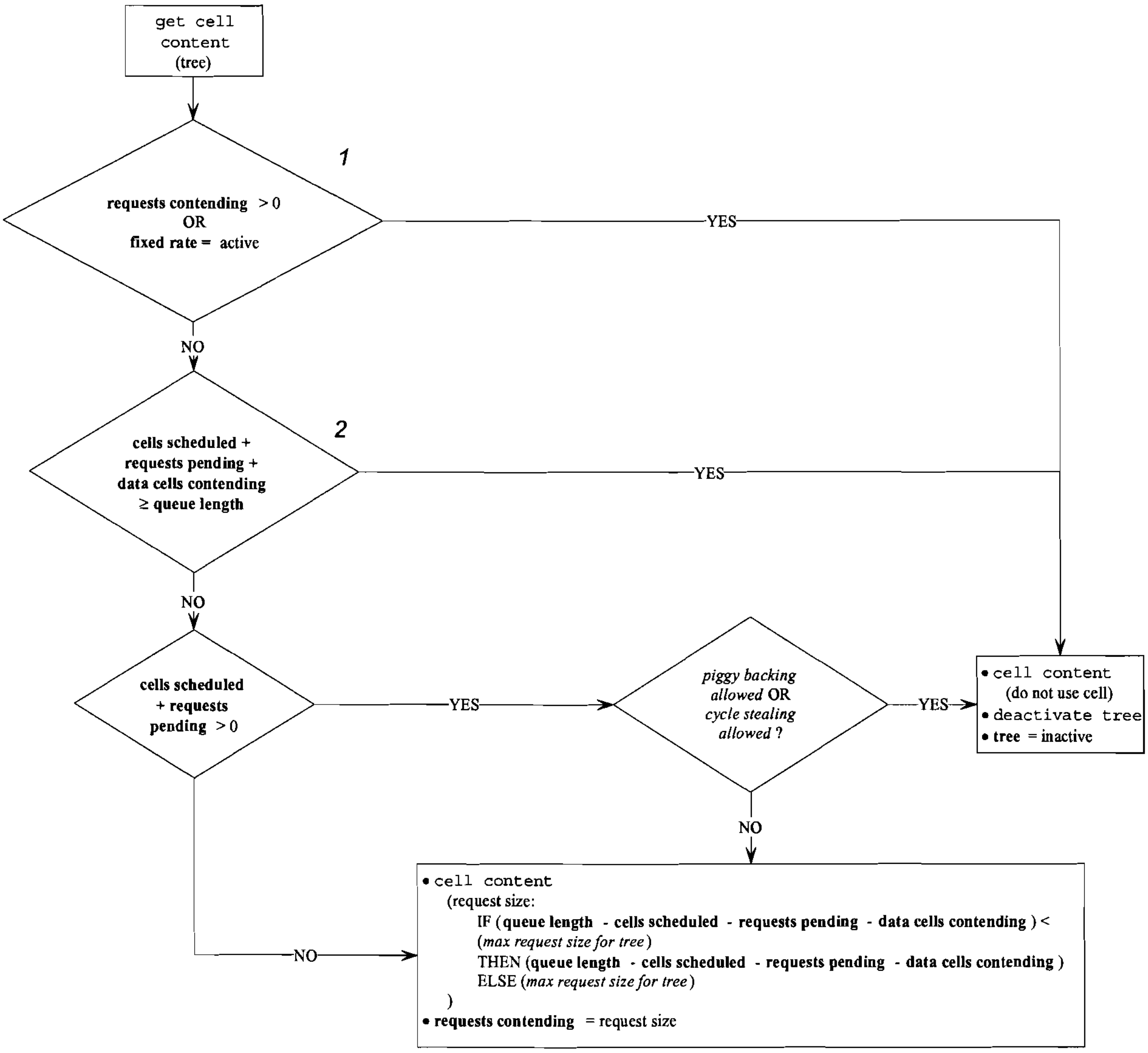


Figure A-6: Flowchart of decisions and actions after a `get cell content` (tree).

Notes concerning the flowchart in Figure A-6:

ad. *1*: If there are requests contending do not use cell. This is done to make sure that requests are not contending in both ALOHA and tree. Otherwise the status variable, **requests contending**, could contain requests that are contending in ALOHA and tree with no possibility to discriminate them. Therefore it is not clear which part of **requests contending** is involved when contention feedback for ALOHA or tree arrives. Adding status variables can solve this, but we thought it would not be interesting enough.

ad. *2*: When there is no need for sending new requests then do not use the tree cell. The **data cell contending** variable in the condition should prevent from requesting more cells than needed.

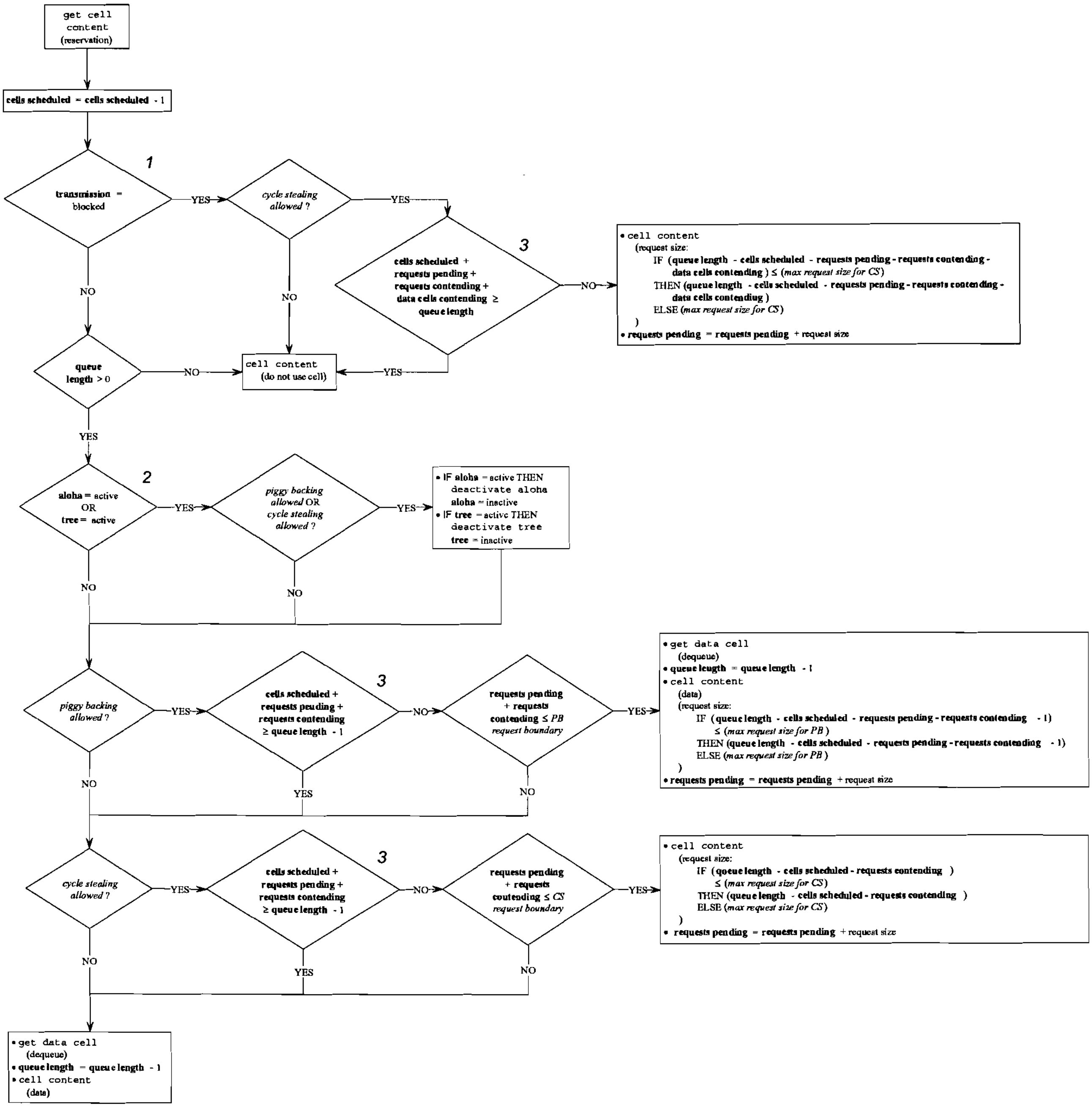


Figure A-7: Flowchart of decisions and actions after a `get cell content` (reservation).

Notes concerning the flowchart in Figure A-7:

ad. *1*: When **transmission** is blocked, it is not permitted to send a data cell. When data is sent in contention and no contention feedback is received yet, sending data in reservation could result in sending a cell twice or changing the order of reception of data cells. However, when allowed, it is possible to use the reservation cell for cycle stealing and request a number of new reservation cells.

ad. *2*: If **aloha** or **tree** is active (not active locked) and piggybacking is allowed, the NTS aloha processor or NTS tree processor is deactivated. This is done to prevent from sending requests in contention and in this way reduce the burden on the contention channel.

ad. *3*: Conditions that check the need for sending new requests.

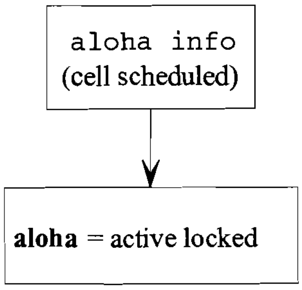


Figure A-8: Flowchart of the action after a `aloha info` (cell scheduled).

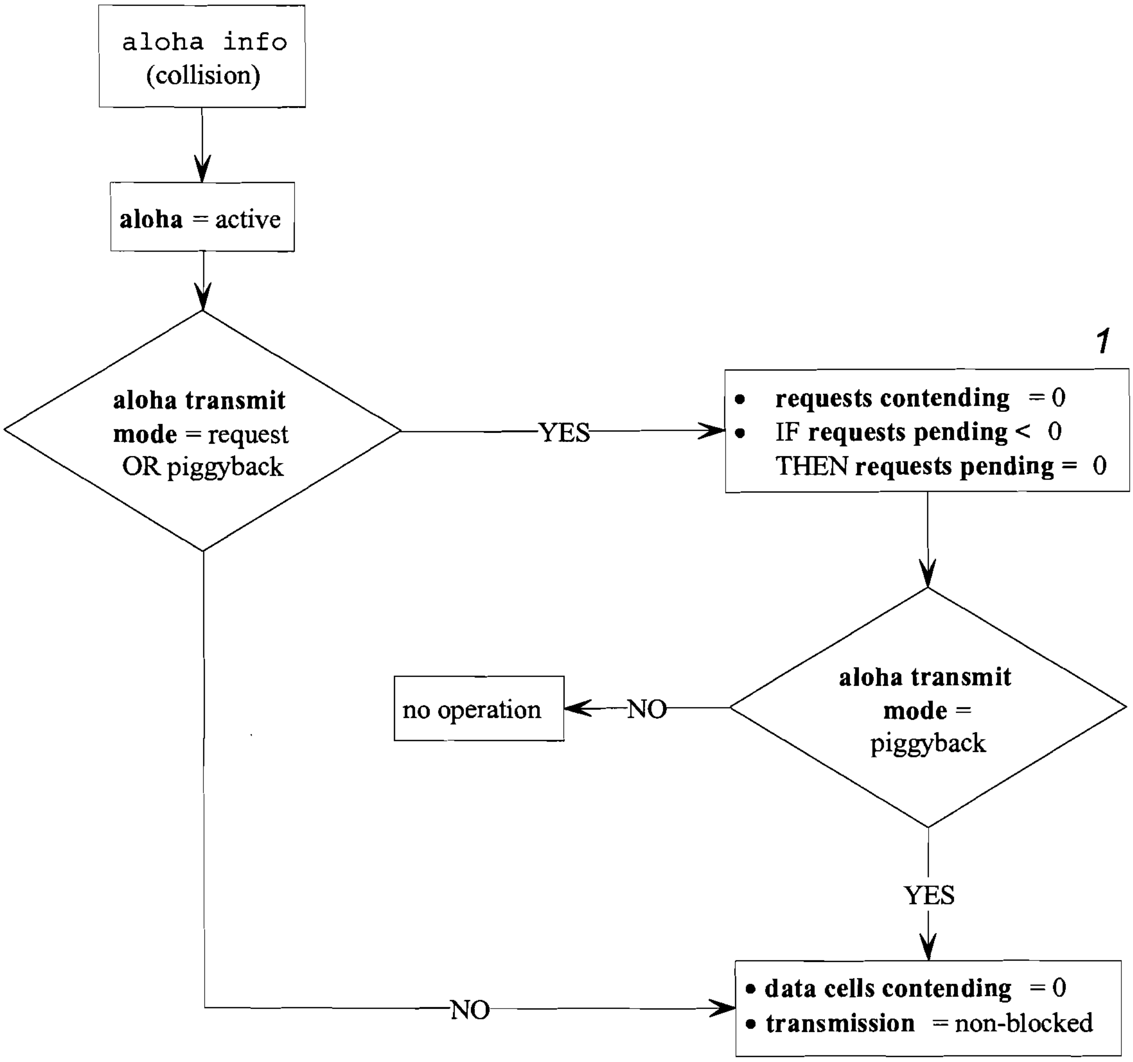


Figure A-9: Flowchart of the decisions and actions after a `aloha info` (collision).

Notes concerning the flowchart in Figure A-9:

ad. *1*: In this block **requests pending** is corrected, when it comes below zero. This situation can occur for two reasons: (1) the HE gives reservation grants that were not requested and (2) reservation grants are received as a result of requests sent in contention before contention feedback for these requests is received. After reception of feedback about a collision it is clear that the latter situation does not apply.

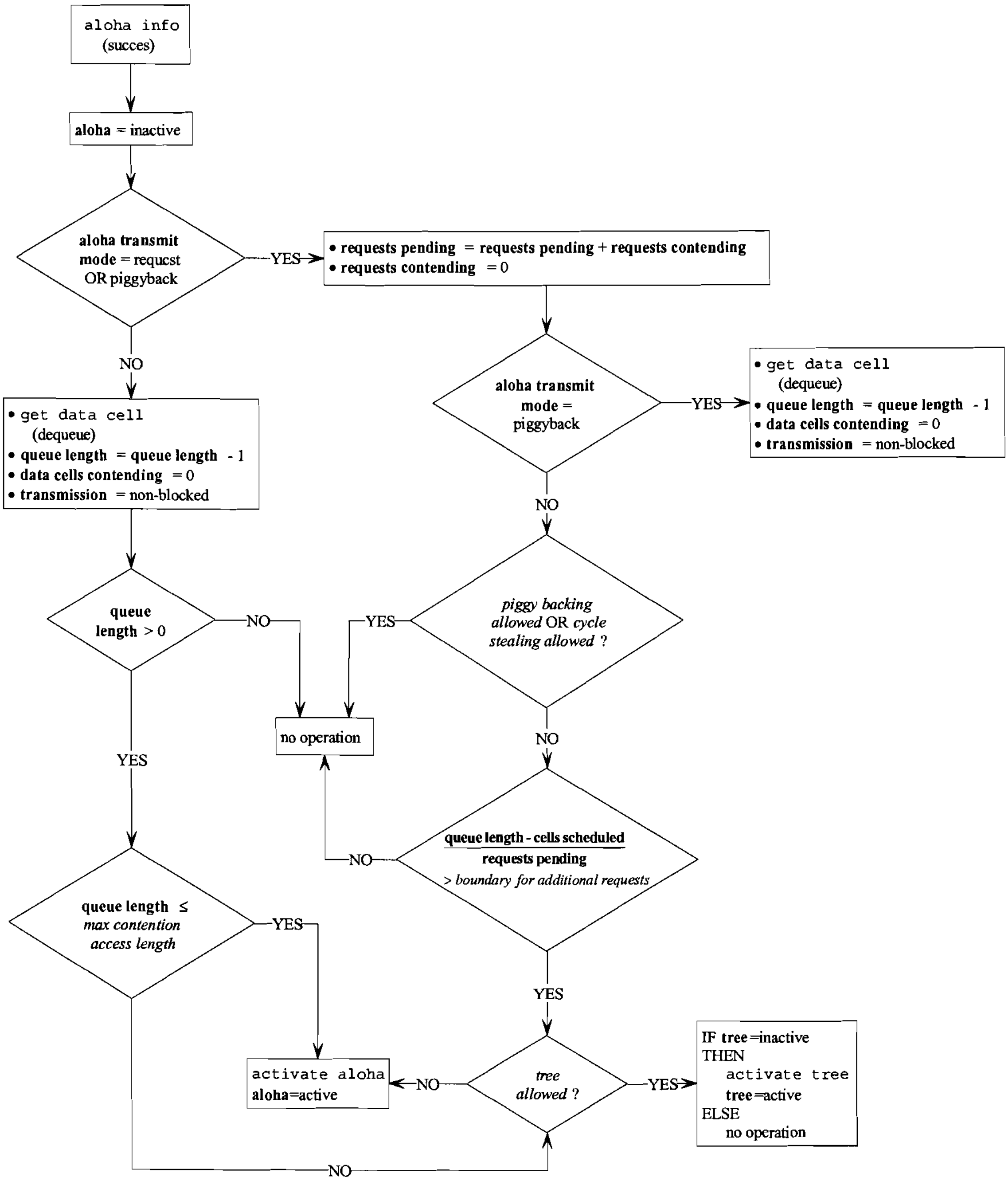


Figure A-10: Flowchart of the decisions and actions after a `aloha info` (success).

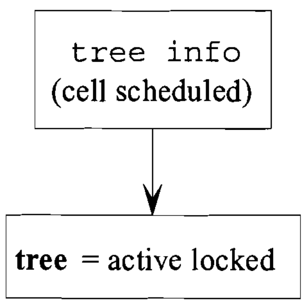


Figure A-11: Flowchart of the action after a `tree info` (cell scheduled).

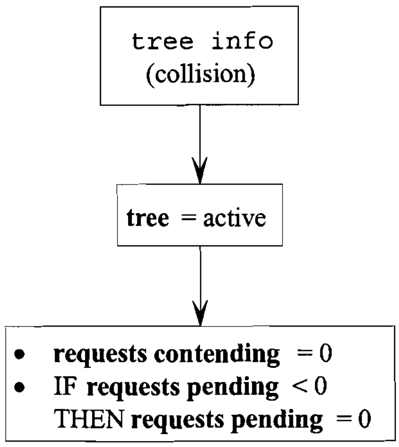


Figure A-12: Flowchart of the decisions and actions after a `tree info` (collision).

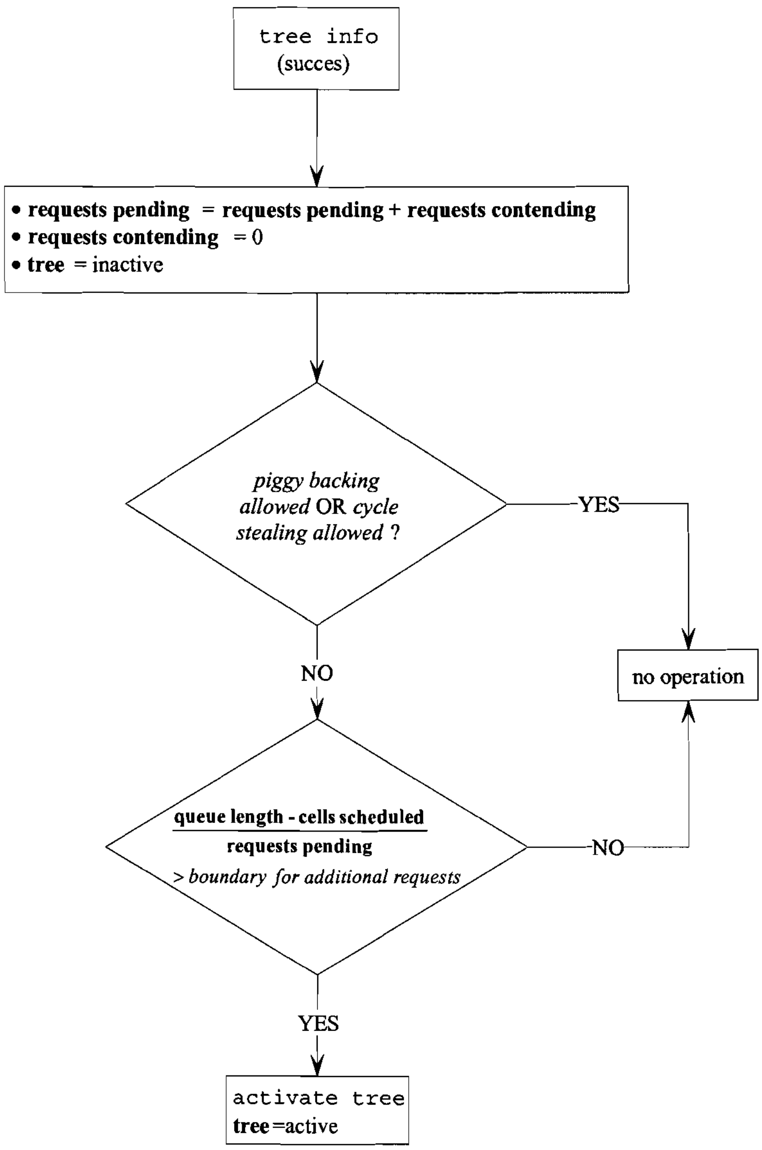


Figure A-13: Flowchart of the decisions and actions after a `tree info` (success).

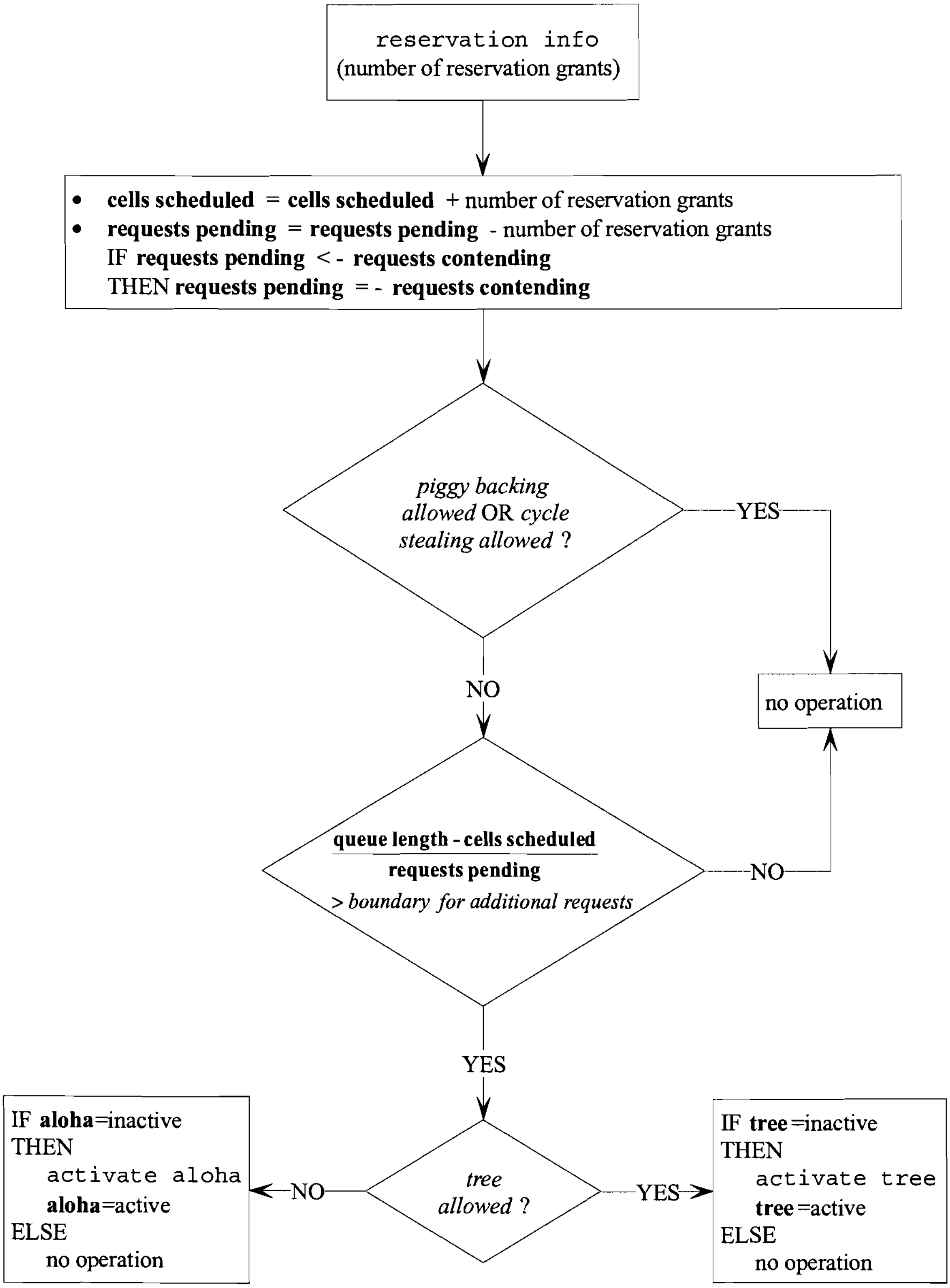


Figure A-14: Flowchart of the decisions and actions after a `reservation info`.