

Dynamic Capacity Allocation for Quality-of-Service Support in IP-Based Satellite Networks

Nicolae Iuoras

Department of Electrical & Computer Engineering
McGill University, Montreal
December 2004



A thesis submitted to McGill University
in partial fulfillment of the requirements of the degree of
Master of Engineering

© Nicolae Iuoras 2004



Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 0-494-12615-9

Our file Notre référence

ISBN: 0-494-12615-9

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

ABSTRACT

Since the Internet has become the de-facto global networking infrastructure, IP-based services have increasingly been used for delivering multimedia applications, including various voice, video and data applications. Such applications generate traffic with different characteristics and consequently, require various levels of service. Service differentiation and end-to-end Quality-of-Service (QoS) provisioning in IP networks thus represent a major preoccupation in both the research community and industry. Recent years have also witnessed considerable interest in delivering IP-based multimedia applications using broadband satellite access (BSA) networks. Such networks rely on multi-beam technologies and a satellite-based return channel in order to provide a viable solution for two-way broadband Internet access to users in areas without adequate terrestrial infrastructure. The efficient management of the return channel transmission resources (capacity) is a key element in reducing the service cost, and thus, in ensuring the viability of BSA for IP multimedia services.

This thesis concentrates on advanced dynamic capacity allocation (DCA) schemes that allow an efficient and fair sharing of the BSA return channel capacity among many user terminals (UTs), by dynamically adapting each UT's capacity assignment to the input traffic variations, while satisfying QoS requirements. DCA performance is evaluated with respect to satellite bandwidth utility (utilization) and QoS performance experienced by user traffic (expressed in terms of packet delay and loss). DCA performance is affected by various elements/parameters, some internal to the DCA scheme and others, external. Relying on simulation, the thesis considers the effects of these elements/parameters on performance, focusing on their optimization in order to balance the DCA performance against the signaling overhead and computational/algorithmic complexity of the DCA scheme. A novel DCA scheme is proposed, capable of providing Differentiated Services (DiffServ) IP QoS support for multimedia applications and high bandwidth utility, while maintaining reduced signaling overhead and computational/algorithmic complexity.

SOMMAIRE

L'Internet est devenu dernièrement le réseau global pré-éminent ; par la suite, les services IP ont donc été de plus en plus utilisés pour la livraison d'applications multimédias, telles que voix, vidéo et données. Ces applications génèrent du trafic à différentes caractéristiques et par conséquent, nécessitent de divers niveaux de service. La différenciation et provision bout à bout de la Qualité de Service (QoS) représentent ainsi une préoccupation majeure dans les milieux de recherche et dans l'industrie. Les dernières années ont aussi témoigné d'un intérêt considérable pour la livraison des applications IP multimédias à travers des réseaux d'accès par satellite à bande large (ASBL). Ces réseaux sont basés sur des technologies à multi-faisceaux et incluent un canal de retour par satellite, afin d'offrir une solution viable pour l'accès aller-retour de bande large aux usagers situés dans des régions sans infrastructure terrestre adéquate. La gestion efficace des ressources de transmission (capacité) du canal de retour est essentielle à la réduction du coût du service, et donc, à la viabilité des ASBL pour les services IP multimédias.

Cette thèse est consacrée à l'étude de schèmes avancés pour l'allocation dynamique de capacité (ADC) capables d'assurer la distribution efficace et équitable de la capacité du canal de retour à de nombreux terminaux de satellite (TS), grâce à l'adaptation dynamique de la capacité allouée à chaque TS aux variations du trafic input, tout en répondant aux exigences QoS. La performance de l'ADC est évaluée par rapport à l'utilisation efficace de la capacité du satellite et à la QoS perçue par l'utilisateur (exprimée en termes de délai et perte de paquets). La performance de l'ADC est affectée par de divers éléments/paramètres, soit internes, soit externes au schème ADC. En utilisant la simulation, la thèse considère les effets de ces éléments/paramètres sur la performance, se concentrant sur leur optimisation afin de balancer la performance de l'ADC contre ses exigences de signalisation et sa complexité algorithmique. Un nouveau schème ADC est proposé, capable d'offrir le support QoS (basé sur les Services Différenciés) aux applications multimédias, tout en utilisant efficacement la capacité du satellite, avec signalisation et complexité algorithmique réduites.

ACKNOWLEDGEMENTS

First and foremost, I would like to sincerely thank my research supervisor, Dr. Tho Le-Ngoc. He unfailingly provided invaluable guidance and support over the entire course of my M. Eng. studies, while also encouraging me to pursue my own ideas, and was never too busy to address my questions or concerns.

I would also like to thank my colleagues, especially Mohamed Ashour and Tallal Elshabrawy, for taking the time to provide help and constructive comments concerning my research in many discussions.

Finally, I wish to thank my parents, Mr. Augustin Iuoras and Mrs. Andrea Iuoras, for their constant love and support.

TABLE OF CONTENTS

ABSTRACT	2
SOMMAIRE	3
ACKNOWLEDGEMENTS	4
Chapter 1 : INTRODUCTION.....	7
1.1. Background	7
1.2. Motivation / rationale	9
1.3. Thesis objectives and contributions	11
1.3.1. Traffic modeling assumptions.....	12
1.3.2. Estimation of the mean traffic arrival rate using an averaging window	13
1.3.3. Network overload probability and time distribution	14
1.3.4. Capacity request/assignment algorithms and signaling overhead.	15
1.3.5. DCA optimization using simulation.....	16
1.3.6. Effects of traffic state switching rate on DCA performance	16
1.4. Thesis logic (flow) summary.....	17
1.5. Thesis organization	18
Chapter 2 : ARCHITECTURES FOR IP QoS-BASED BSA NETWORKS	21
2.1. BSA network architectures and trends	21
2.1.1. Overall reference architecture	21
2.1.2. Upstream uplink access using MF-TDMA	23
2.1.3. Multi-beam satellite architectures and inter-beam switching.....	25
2.1.3.1. Non-regenerative payload: frequency switching.....	27
2.1.3.2. Regenerative payload: on-board switching	28
2.1.4. Dynamic capacity allocation and scheduling	29
2.2. IP QoS architectures.....	32
2.2.1. End-to-end QoS.....	32
2.2.2. Integrated Services (IntServ).....	34
2.2.3. Differentiated Services (DiffServ)	35
2.3. DiffServ-capable scheduler and UT architectures.....	37
2.3.1. Scheduler.....	37
2.3.2. User terminal	42
Chapter 3 : A DYNAMIC CAPACITY ALLOCATION SCHEME FOR	
DiffServ QoS SUPPORT	47
3.1. QoS performance measure	47
3.2. Background and previous work.....	48
3.3. Input traffic modeling.....	52
3.4. Moving-average estimation of the mean traffic arrival rate.....	55
3.5. Network overload probability and mean overload time	57
3.5.1. Network overload probability	58
3.5.2. Mean network overload time.....	59
3.6. Proposed DCA scheme.....	62
3.6.1. Case 1 (C1): Frame-based requests with known traffic arrival rates (as baseline).....	63

3.6.2.	Case 2 (C2): Frame-based requests with estimated traffic arrival rates	68
3.6.3.	Case 3 (C3): Reduced request and assignment frequency (signaling) and simplified scheduling algorithm.....	69
Chapter 4	: PERFORMANCE EVALUATION	72
4.1.	Simulation model description.....	72
4.1.1.	Model configuration	73
4.1.2.	Simulation scenarios	75
4.1.2.1.	Network loading	75
4.1.2.2.	SRD vs. LRD traffic	76
4.1.2.3.	Moving-average sliding window size	77
4.1.2.4.	Request transmission threshold.....	77
4.2.	Results and discussions	78
4.2.1.	Comparison of DCA cases	78
4.2.2.	QoS performance ($\Pr\{Q>n\}$)	82
4.2.2.1.	Effects of the moving-average sliding window size	85
4.2.2.2.	Effects of the request transmission threshold.....	87
4.2.3.	Network overload	90
4.2.4.	Effects of traffic state switching rate on DCA performance	92
Chapter 5	: SUMMARY AND CONCLUSIONS	96
REFERENCES	101

Chapter 1 : INTRODUCTION

1.1. Background

The Internet has become the de-facto global networking infrastructure, built primarily around terrestrial core and access networks using various transport technologies; the Internet Protocol (IP) is the common denominator in all these networks. Due to the ubiquitous use of IP in terrestrial networks, IP-based services have been increasingly used for delivering multimedia applications, including various voice, video and data applications. Such applications generate traffic with different characteristics (e.g., burstiness) and thus, require different levels of service. Real-time applications like voice and videoconferencing, for example, are delay and jitter sensitive; data applications, on the other hand, are more tolerant to delay and jitter but may be more sensitive to packet losses (e.g., banking transactions).

Due to the various service requirements of multimedia applications, service differentiation and end-to-end Quality-of-Service (QoS) guarantees are increasingly being demanded by the users of IP networks and represent a major preoccupation in both the research community and industry. The original IP [1] has no notion of QoS and only provides best-effort delivery of IP packets (datagrams). Two key architectures/models have since been proposed under the auspices of the Internet Engineering Task Force (IETF) in order to provide IP QoS support: Integrated Services (IntServ) [2] and Differentiated Services (DiffServ) [3] [4]. The IntServ architecture allows per-flow end-to-end connections (virtual circuits) with pre-defined bandwidth to be established. All network nodes (routers) along the end-to-end path must reserve forwarding resources (buffer space and bandwidth) and maintain state information for each individual IP flow; this increases equipment complexity and raises scalability concerns in large networks. In order to circumvent these shortcomings, the DiffServ architecture has been proposed. It specifies the forwarding treatment, called per-hop behavior (PHB), that IP packets should receive at network nodes depending on their service class. Five service classes (PHBs) in addition to the

best-effort class have been defined. PHBs apply to traffic aggregates, not individual flows, so the DiffServ architecture leads to simpler, more scalable implementation than IntServ. DiffServ has therefore become the architecture of choice for QoS support in IP networks. In order to provide end-to-end QoS based on the DiffServ architecture, each network along a connection's end-to-end path must support the DiffServ PHBs by implementing the appropriate forwarding mechanisms in all the network nodes; in addition, end-to-end protocols are needed for session and QoS signaling.

In recent years, there has been considerable interest in delivering multimedia applications using broadband satellite access (BSA) networks [5]. The most obvious advantage of BSA over terrestrial access technologies (e.g., cable, DSL) is its ability to provide connectivity in remote areas without adequate terrestrial infrastructure, thereby offering an economically viable solution for "last mile" broadband access by providing a so-called "instant infrastructure". In addition, the wide-area coverage of a satellite allows long-distance communication with only one hop, thus avoiding the often-unpredictable delays resulting from routing and congestion in terrestrial networks. Finally, the broadcast nature of the satellite medium is ideal for efficiently providing multicast and broadcast services to large user populations.

Since many multimedia applications are interactive (conversational), they require bi-directional (two-way) communication channels. Until recently, the deployment of BSA networks for these applications has been somewhat hindered by the lack of a satellite-based return channel to transport the user traffic. The last few years have witnessed the development of such return-channel technologies. In Europe, a return channel specification has been defined under the auspices of the European Telecommunications Standards Institute (ETSI) in the Digital Video Broadcasting – Return Channel via Satellite (DVB-RCS) open standard [6]. The standard has been used by SES-Astra for the deployment of the BroadBand Interactive (BBI) system – the first system offering satellite interactive multimedia services on a commercial basis. While DVB-RCS is gaining acceptance all over the world and is being used by many satellite operators (e.g., Telesat, Eutelsat, JSAT) for

experimental systems (demonstrators) and even for operational systems, other return channels have been developed based on proprietary technologies (especially in the USA, by companies such as Gilat and Hughes Network Systems).

The high cost of interactive services via satellite (compared to terrestrial technologies) has also hindered the deployment of BSA networks, but this is also about to change. The service cost is dominated by two main elements, namely the cost of user terminals (UTs) and the scarcity of satellite resources (bandwidth, buffering and computational resources). High-gain multi-beam antennas at the satellite have made possible the use of small antennas and low-power transmitters at the UTs, which, combined with more efficient modulation/coding schemes, create the premise for UT cost reduction to levels comparable to those offered by other technologies (e.g., terrestrial wireless). Multi-beam technology also provides increased system capacity for a given bandwidth through frequency reuse between beams; this allows a larger number of users to be supported, leading to further reduction of the per-user service cost by the principle of economies of scale. The service cost can be reduced even further through the efficient utilization of the available satellite bandwidth, by using advanced capacity allocation schemes.

1.2. Motivation / rationale

To capitalize on IP's ever-increasing popularity, BSA networks should seamlessly interoperate with terrestrial IP networks as part of the global communication infrastructure. Consequently, BSA networks should be compatible with IP-based technologies, architectures and protocols capable of providing service differentiation and QoS guarantees. For the reasons described in Section 1.1, DiffServ is the preferred architecture for IP QoS provisioning capable of meeting the requirements of multimedia applications.

In BSA networks, the return link capacity is subject to contention among many UTs. The contention is resolved by the Medium Access Control (MAC) protocol, which governs the assignment of capacity to the UTs. The performance of the

MAC protocol is judged using two sometimes-conflicting criteria:

- The utility (utilization) of the available satellite transmission bandwidth. A high utilization is desired in order to increase the number of supported users, thus reducing the per-user bandwidth cost.
- The QoS performance offered to users, typically defined in terms of performance metrics including the packet delay, delay variation (jitter) and packet loss. Acceptable values for these metrics depend on the application; a voice-over-IP (VoIP) application, for example, requires low delay and is very sensitive to jitter, while a file transfer is rather insensitive to delay and jitter but requires low packet loss.

One option is to make each UT's capacity assignment static (fixed). This method is simple and provides good link utility with applications generating traffic with constant or nearly constant rates (e.g., audio/video streaming). Unfortunately, most multimedia applications, such as web browsing, file transfers and videoconferencing, generate variable or even bursty traffic. In this case, the fixed assignment scheme would lead to either low utility but good QoS performance, if the assigned capacity is close to the application's peak transmission rate, or high utility but inadequate QoS performance due to queue buildups (increased delays) and/or overflows (packet losses), if the assigned capacity is close to the application's mean rate. Neither of these outcomes is acceptable, therefore Dynamic Capacity Allocation (DCA) schemes are required. DCA schemes rely on *capacity requests* from the UTs reflecting their needs, and corresponding *capacity assignments* from a (typically centralized) scheduler. This process can effectively improve resource utility while maintaining adequate QoS performance for the user traffic, by dynamically adapting the capacity assignments to the UTs' input traffic variations. With regard to QoS performance, it is noted that most multimedia applications can cope with the scheduling delay (latency) introduced by DCA, i.e., the propagation time of the capacity requests and assignments over the satellite link and the processing time within the UTs and the scheduler. This scheduling delay is called the DCA cycle time, or request-to-assignment time. Compared to the fixed assignment strategy, DCA comes at the price of increased

signaling overhead and complex algorithms for calculating capacity requests and assignments. The request and assignment signaling requires precious satellite transmission bandwidth, which could otherwise be used for user traffic; this increases the service cost per unit bandwidth. The complexity of the algorithms requires additional computational resources, which increase the cost of the equipment (UTs and scheduler). Understandably, satellite network operators wish to achieve the high resource utility offered by DCA, but with low signaling overhead and simple request and assignment algorithms. On the other hand, users are interested in good QoS performance for their applications at low cost. The variations of the traffic generated by some of these applications may require frequent updates of the capacity request and assignment amounts, in order to satisfy QoS requirements; these frequent updates would increase the amount of signaling. A more responsive DCA scheme (i.e., one that more closely follows the input traffic variations) uses more signaling to update the capacity requests and assignments but may provide better performance (in terms of satellite bandwidth utility and user QoS performance), because the assigned capacity more accurately reflects each UT's input traffic variations. A *tradeoff* is therefore required between the signaling overhead, the complexity of a DCA scheme and the scheme's performance. An incremental performance improvement may not justify a significant increase in the amount of signaling and/or the scheme's complexity; a *balance* must therefore be established between these three elements.

1.3. Thesis objectives and contributions

The discussion in Section 1.2 indicates that the efficient delivery of multimedia applications over BSA networks requires the use of DCA. The main research theme/objective of this thesis is to propose a novel DCA scheme that is efficient with regard to satellite bandwidth utility and satisfies user QoS requirements, while maintaining reduced signaling overhead and computational complexity. User QoS requirements are expressed in terms of QoS performance parameters, primarily packet delay and loss at the UT.

DCA performance depends on a number of elements/parameters, some of which

are internal to the DCA scheme and others, external. In the subsections below, we identify these elements/parameters and discuss various alternatives/options regarding their choice, the expected performance effects of each of these alternatives and the associated research issues considered in the thesis. The investigation and resolution of these issues constitute the main contributions of the thesis.

The design and performance of the DCA scheme depend to a great extent on the characteristics of the aggregate user traffic in the BSA network. Appropriate input traffic models are therefore required, and the DCA design should be tailored to these models. Traffic modeling is not the focus of this thesis, however; results from the literature and assumptions regarding the typical operational environment of the UTs are therefore used to select a traffic model and justify its validity. This creates the premise for the development of the DCA scheme and the evaluation and optimization of its performance using mainly simulation, but also analysis where appropriate.

1.3.1. Traffic modeling assumptions

The selected input traffic model is based on two-state switched Poisson traffic sources. The packet inter-arrival times in both states are exponentially distributed with different means; hence the arrival process in each state is a Poisson process. The state with the higher mean packet arrival rate (lower mean inter-arrival time) is called the “high” state, and the other, the “low” state. Each traffic source switches (toggles) between its two states; the traffic sources operate independently of each other, i.e., their state switching points are not synchronized. The state sojourn times are specified by a probability distribution that is either exponential, in which case the resulting traffic is said to exhibit short-range dependence (SRD), or heavy-tailed (e.g., Pareto), in which case the traffic is said to exhibit long-range dependence (LRD). The SRD and LRD properties refer to the relative decay rates of the traffic’s time autocorrelation function, with implications on queuing performance: a queuing system with LRD input traffic (i.e., slower decay rate) generally results in longer queues (and thus, increased

packet queuing delays and losses) than with SRD traffic. A heavy-tailed state sojourn time distribution implies that for a given mean, there is a higher probability of obtaining longer sojourn times than in the case of distributions that are not heavy-tailed (e.g., exponential). The mean state sojourn times are set greater than the DCA cycle time defined in Section 1.2, so that each traffic source can be assumed to remain in the same state for the entire duration of a DCA cycle. Consequently, the traffic within the DCA cycle behaves as simple (non-switched) Poisson traffic, which is analytically tractable, and the capacity allocation can be based on Poisson traffic's only parameter, its mean arrival rate, estimated (measured) as described below.

1.3.2. Estimation of the mean traffic arrival rate using an averaging window

As noted in Section 1.2, DCA is based on the dynamic adaptation of the capacity assigned to each UT to the UT's input traffic variations; these variations must therefore be measured in order to determine the amount of capacity required. Since the traffic is modeled by two-state switched Poisson sources as described above, the variations are caused both by the stochastic nature of the Poisson traffic arrival process in each state, and by each source's state switching. The latter contributes more than the former to the traffic variations, because there is generally a significant difference between the mean traffic arrival rates in the two states. A method is therefore required to estimate (measure) the mean traffic arrival rate, which changes with the state; this can be accomplished using a window-based moving time-average of packet arrivals. The size of the averaging window is an adjustable parameter affecting the estimated mean, and thus, the DCA performance. With a smaller window, the moving average follows the instantaneous traffic arrival rate more closely, which implies high variation in the estimated mean. As the window size is increased, the estimated mean becomes smoother, i.e., the estimate is of better quality, possibly improving DCA performance; the downside is that the estimate responds more slowly to the traffic source's state switching point, with the opposite effect of worsening DCA performance. Given this tradeoff between the estimate's quality and its

responsiveness to traffic variations, the optimization of the window size to maximize DCA performance is a significant research issue.

1.3.3. Network overload probability and time distribution

As stated in Section 1.3.1, the input traffic in the BSA network is modeled by two-state switched Poisson traffic sources operating independently. If enough of these sources are simultaneously in their “high” state, the aggregate traffic arrival rate to the network may momentarily exceed the total transmission capacity (although the aggregate *mean* traffic arrival rate does not), resulting in a *network overload*, until some of the traffic sources in the “high” state switch to the “low” state. During network overloads, the UTs’ queues become unstable (build up) because the traffic arrival rates to the queues exceed the queue service rates, resulting in reduced QoS performance (i.e., increased queuing delays and/or packet losses). Consequently, the overloads’ occurrence frequency and duration both affect the QoS performance; overloads should only occur rarely (i.e., the *overload probability* should be low) and their duration should be short, in order to minimize their performance impact. The overload duration is related to the state sojourn times of the traffic sources: longer sojourn times lead to longer overload times, because more time elapses before a traffic source in the “high” state switches to the “low” state. As described in Section 1.3.1, LRD traffic, with its heavy-tailed sojourn time distribution, generally exhibits longer sojourn times with higher probabilities than SRD traffic for the same mean sojourn time. The network overload times are thus presumably longer with LRD input traffic; to determine the difference, the overload times with both traffic types must be compared. A study of the network overload time (more specifically, its mean and probability distribution) is thus a worthwhile research issue. The reduced QoS performance resulting from the prolonged overload times with LRD traffic can be counteracted by lowering the network traffic load.

1.3.4. Capacity request/assignment algorithms and signaling overhead

A DCA scheme consists of the algorithms and equations used to calculate the capacity request and assignment amounts. The request amounts could be based on the number of new packet arrivals to the UT since the last capacity request. Alternatively, the request amounts could reflect estimates of the capacity required in order to achieve some target QoS requirement; an analytical queuing model could be used to calculate these estimates, depending on the mean traffic arrival rate. In principle, this second request strategy could result in better DCA performance than the first because the target QoS level is taken into account in the capacity estimates. In either case, the capacity request and assignment algorithms must be matched. As noted in Section 1.2, the computational complexity of these algorithms should be weighed against the resulting DCA performance: an incremental performance improvement may not justify a significant increase in complexity.

In addition to the performance-complexity tradeoff described above, another significant research issue involves the performance-signaling tradeoff: recall from Section 1.2 that a DCA scheme that is more responsive to input traffic variations (i.e., that more closely follows these variations) uses more signaling to update the capacity requests and assignments, but may provide better performance because the assigned capacity more accurately reflects each UT's traffic conditions. Now, recall from the discussion of traffic modeling in Section 1.3.1 that the mean state sojourn times of the switched Poisson input traffic sources are set greater than the DCA cycle time, so that each traffic source can be assumed to remain in the same state for the entire duration of a DCA cycle. Consequently, the traffic within the DCA cycle behaves as simple (non-switched) Poisson traffic, which is analytically tractable, and the capacity allocation can depend on the traffic's mean arrival rate, measured using a moving time-average of packet arrivals, as described in Section 1.3.2. To reduce signaling, the capacity requests should only be updated when this mean arrival rate changes significantly; this can be accomplished by calculating the relative change in the measured mean arrival rate since the last capacity request was transmitted, and only transmitting a new

request if this relative change exceeds an adjustable parameter called *request transmission threshold*. The optimization of this threshold is a significant research issue: if it is set too high, the input traffic source's state switching points may be "missed" (overlooked) by the DCA scheme, leading to reduced QoS performance; conversely, if the threshold is too low, an excessive number of capacity requests would be transmitted, increasing the signaling overhead without necessarily improving the QoS performance significantly. The optimization of the request transmission threshold is related to the optimization of the traffic averaging window size described in Section 1.3.2, since both of these parameters affect the DCA scheme's responsiveness to input traffic variations.

1.3.5. DCA optimization using simulation

A simulation model is developed in order to evaluate the DCA performance and to optimize the DCA elements/parameters described in the previous sections. In particular, this optimization includes selecting the capacity request and assignment algorithms (in view of the performance-complexity tradeoff), the traffic averaging window size (in view of the quality-responsiveness tradeoff defined in Section 1.3.2), and the request transmission threshold (in view of the performance-signaling tradeoff).

1.3.6. Effects of traffic state switching rate on DCA performance

In all previous sections, it was assumed that the mean state sojourn times of the switched Poisson input traffic sources are greater than the DCA cycle time; this is referred to as *slow-switching* traffic. This assumption influenced the design of the proposed DCA scheme. Simulation is now used to evaluate the DCA performance with *fast-switching* input traffic (i.e., when the mean state sojourn times of the traffic sources are *smaller* than the DCA cycle time), and thus, to determine whether the DCA scheme must be adjusted for operation with fast-switching traffic, since it is no longer possible to make the simplifying assumption that each traffic source remains in the same state for the entire duration of a DCA

cycle, as in Section 1.3.1.

1.4. Thesis logic (flow) summary

Having identified in Section 1.3 the key elements/parameters affecting DCA performance and the associated research issues to be considered in the thesis, the logic (flow) of the thesis, established in the previous sections, can be summarized as follows.

The multimedia applications envisioned for BSA have different QoS requirements. Furthermore, the traffic generated by these applications is in general bursty; consequently, in the satellite environment, DCA schemes are required in order to provide high satellite link utility while meeting QoS requirements. A DCA scheme that is efficient with regard to link utility may be costly in terms of the required signaling overhead and the complexity of the algorithms used for capacity request and assignment calculation. Excessive signaling, while beneficial to DCA responsiveness, requires satellite transmission bandwidth that could otherwise be used for user traffic, thus reducing the link utility and defeating the purpose of DCA. The challenge is therefore to develop a DCA scheme capable of providing high link utility with low signaling overhead, while maintaining adequate responsiveness in order to satisfy the users' QoS requirements, and low algorithmic complexity in order to reduce computational requirements and therefore, hardware cost. This challenge can be met through careful consideration and optimization of various DCA elements, notably the capacity request and assignment calculation algorithms and the request frequency. DCA performance depends on the assumptions concerning the modeling of the input traffic in the BSA network. Based on a literature survey of multimedia traffic modeling and on the specifics of the BSA environment, a two-state switched Poisson model is considered adequate for the UT's aggregate input traffic. Such a model can exhibit SRD or LRD behavior, depending on the probability distribution used to describe the state sojourn times. The state switching is generally considered to be slower than the DCA cycle time (i.e., slow-switching traffic), therefore the traffic within a DCA cycle behaves as

Poisson traffic. The capacity requests can thus be calculated based on estimates of the mean traffic arrival rate; window-based time averaging can be used to this end, with an adjustable window size. Furthermore, to reduce signaling, capacity requests need only be sent when there are significant variations in the estimated mean (as compared to an adjustable request threshold). Both the averaging window size and the request threshold should be carefully selected, in order to maintain adequate DCA responsiveness (consistent with meeting the QoS performance requirements) while reducing the signaling overhead.

The state switching of the input traffic sources can lead to network overload situations in which the aggregate traffic arrival rate to the BSA network momentarily exceeds the total transmission capacity, with detrimental effects on QoS performance. Due to its longer state sojourn times (as compared to SRD traffic), LRD traffic can prolong these overload situations; to determine the extent of this prolongation, the overload times with both traffic types must be compared. The reduced QoS performance resulting from the prolonged overload times can be counteracted by lowering the network traffic load.

In cases when the state switching of the input traffic sources is faster than the DCA cycle time (i.e., fast-switching traffic), the sources can no longer be assumed to remain in the same state during a DCA cycle as with slow-switching traffic. Since this assumption influenced the design of the proposed DCA scheme, it is necessary to evaluate the DCA performance with fast-switching traffic in order to determine whether the DCA scheme must be modified/adjusted.

1.5. Thesis organization

The remainder of this thesis is organized as follows.

In Chapter 2, we provide the background/context and reference system architecture necessary for presenting our DCA-related research in subsequent thesis chapters. We begin with a survey/summary of previous work in BSA network architectures and current architectural trends, including the DCA concept. Next, we describe the current state of the two IP QoS architectures in terrestrial networks, IntServ and DiffServ, noting that achieving end-to-end IP

QoS requires cooperation between multiple networks (since the endpoints of an IP connection are often located in different networks) and between multiple protocol layers within the same network; the scope of the research presented in this thesis is limited to MAC-layer QoS provisioning in the BSA network segment of an end-to-end IP connection. Finally, we describe our proposed architectures for a DiffServ-capable scheduler and UT.

In Chapter 3, we present our proposed DCA scheme, which aims to provide high satellite bandwidth utility and satisfy user QoS requirements, while maintaining reduced signaling overhead and algorithmic complexity. We begin by defining an appropriate QoS performance measure, i.e., one that can be used to derive the basic QoS parameters of packet delay and loss. We then provide a survey/summary of previous work on DCA schemes for satellite networks. Next, we present the input traffic model used in the BSA network, justifying its validity using results from the literature and assumptions regarding the typical operational environment of the UTs. We continue by describing the moving-average estimator of the mean traffic arrival rate, a key DCA input. Next, we describe the network overload situation and its expected negative performance impact, setting the stage for our subsequent derivation of the network overload probability and mean overload time. Finally, we present our DCA scheme. A key component of the scheme is an analytical expression for the capacity required by each UT to achieve a target QoS for the user traffic; this expression is derived using an analytical queuing model based on the previously selected input traffic model. Our DCA scheme is presented in the form of three *study cases*, each in itself a self-contained, operational DCA scheme. These three cases represent a logical evolution of the capacity request and assignment algorithms, allowing us to evaluate and compare multiple DCA design options that are in the end consolidated under the third (final) case. Thus, the first two cases are intended only as intermediate steps towards the final case. Our subsequent simulation results in Chapter 4 show that this final case represents the best compromise (tradeoff) between QoS performance, signaling overhead and algorithmic complexity.

In Chapter 4, we use mainly simulation (but also some analytical) results to evaluate the performance of the proposed DCA scheme. Simulation is used for the optimization of the DCA elements/parameters described in Section 1.3; this optimization includes the selection of the capacity request and assignment algorithms (in view of the performance-complexity tradeoff), the traffic averaging window size (in view of the quality-responsiveness tradeoff defined in Section 1.3.2), and of the request transmission threshold (in view of the performance-signaling tradeoff). We begin by describing the simulation model's configuration and selecting appropriate simulation scenarios to evaluate the desired aspects of the DCA performance and facilitate the aforementioned optimization. Next, we present and analyze the simulation results (comparing them to analytical results where appropriate), drawing conclusions regarding the DCA optimization. Finally, we consider the case of fast-switching input traffic, as opposed to the slow-switching traffic considered hitherto: simulation is used to determine how fast-switching traffic affects the DCA performance, and thus, whether the proposed DCA scheme must be adjusted for operation with fast-switching traffic. In Chapter 5, we summarize the work performed in the thesis and briefly restate the conclusions of this work. These conclusions reflect the resolution of the research issues considered in the thesis, i.e., those defined in Section 1.3.

Chapter 2 : ARCHITECTURES FOR IP QoS-BASED BSA NETWORKS

In this chapter, we provide the background/context and reference system architecture necessary for presenting our DCA-related research in subsequent thesis chapters. In Section 2.1 we give an overview of previous work in BSA network architectures and current architectural trends, including the DCA concept. In Section 2.2 we describe end-to-end IP QoS and the current state of the two QoS architectures in terrestrial IP networks, IntServ and DiffServ. Finally, in Section 2.3 we present our proposed architectures for a DiffServ-based scheduler and UT.

2.1. *BSA network architectures and trends*

2.1.1. Overall reference architecture

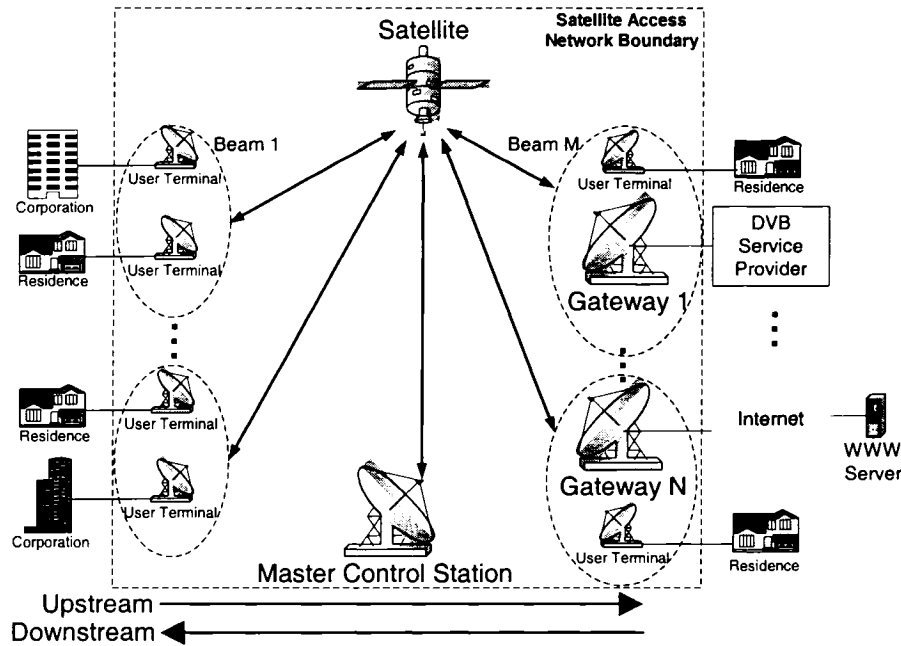


Figure 1: Overall BSA network architecture

Figure 1 shows the overall architecture of a typical bi-directional multi-beam BSA network consisting of UTs, gateways, a satellite and a Master Control Station

(MCS). We use this architecture as a *reference* to provide a context/platform for describing our DCA-related research in the subsequent chapters of the thesis. The reference architecture is based on the BSA architecture described in [5]. We define the *downstream* direction as the direction of traffic flowing from the gateways towards the UTs; conversely, the *upstream* direction refers to traffic flowing from the UTs towards the gateways. The downstream and upstream directions are alternatively referred to as *forward link* and *return link*, respectively. We also define the *downlink* to refer to transmissions from the satellite to terrestrial receivers (e.g., the UTs); downlink transmissions are of a broadcast nature. Conversely, the *uplink* refers to transmissions from terrestrial transmitters to the satellite. The roles of the various components in the BSA network are summarized below.

- The *UTs* are located at the user premises (typically a corporation or residence). They communicate with the gateways and/or other UTs over the satellite link, allowing the users of the BSA network to access various services offered by service providers (SPs). The UT is typically equipped with a standard Local Area Network (LAN) interface (most commonly Ethernet), allowing the user to connect any type of LAN equipment (e.g., PCs, IP phones).
- The *gateways* provide the interface between the BSA network and terrestrial SP networks. In Figure 1, for instance, a Digital Video Broadcasting (DVB) SP is connected to gateway 1, supplying television service; an Internet SP is connected to gateway N, allowing access to Internet resources such as World Wide Web (WWW) servers. In principle, a BSA gateway can provide an interface for any type of terrestrial network (e.g., IP-based Internet, ATM) by using the appropriate protocol translation procedures; in this thesis, however, the emphasis is on IP-based networks.
- The basic function of the *satellite* is to broadcast on the downlink any traffic received on the uplink. Depending on the sophistication of its payload architecture, the satellite may also perform other functions such as packet switching or protocol translation as it forwards the traffic, or it may simply

forward the traffic as is (see Section 2.1.3 for a more detailed discussion). In BSA applications, the satellite is in geosynchronous orbit. The satellite hop delay is defined as the propagation delay for a single ground-satellite-ground hop; for geosynchronous satellites it is approximately 270 ms.

- The *Master Control Station* (MCS) provides the overall control/management functions of the satellite network, including resource control (capacity allocation), transmission scheduling and UT authentication and admission.

Satellite networks may offer two types of connectivity depending on the payload architecture: UT-gateway and UT-UT, respectively called *star connectivity* and *mesh connectivity*. Mesh connectivity typically relies on on-board switching and also implies support for star connectivity, but the converse is not necessarily true. For mesh connectivity, the uplink/downlink nomenclature is more appropriate than the upstream/downstream or forward link/return link nomenclature defined above because the gateways are not communication endpoints. Since BSA networks are fundamentally *access* (not backbone) networks, the BSA service model assumes that most communication occurs between the BSA users (i.e., the UTs) and SPs connected to the gateways; hence, BSA networks are associated mostly with star connectivity. Mesh connectivity can be emulated in star-type networks by using two UT-gateway satellite hops (source UT \rightarrow gateway \rightarrow destination UT). While this approach is not as efficient as true mesh connectivity and introduces an additional traffic delay due to the extra satellite hop, it nevertheless provides more flexibility for star-type BSA networks without the increased system complexity of mesh-type networks.

2.1.2. Upstream uplink access using MF-TDMA

In BSA networks, the downstream uplink aggregate traffic flow from a gateway to the satellite is considered to be smooth (i.e., the peak-to-average ratio is close to 1) because a gateway transmits the aggregate traffic destined for many UTs, thus benefiting from the smoothing effects of statistical multiplexing. Similarly, the upstream downlink traffic flow from the satellite to the gateway is also considered to be smooth because the gateway receives the aggregate traffic from many UTs.

Hence, static (fixed) or, at most, slowly varying capacity allocation is used in these two cases. On the upstream uplink, however, the available transmission capacity is subject to contention among many UTs. The traffic from each UT tends to be bursty because it represents a much smaller degree of aggregation (i.e., only the traffic generated by the applications of the local UT users) than the traffic flowing to/from a gateway. In addition, there are typically tens or hundreds of thousands of UTs in a BSA network. For these reasons, an efficient MAC is required in order to efficiently and fairly share the available upstream uplink transmission capacity among the UTs. MAC scheduling using DCA is at the core of this thesis; hence, many of the architectural aspects described in the present chapter are regarded from the MAC scheduling point of view.

Multiple Frequency Time Division Multiple Access (MF-TDMA) [7] is widely used as a primary access scheme for the upstream uplink of satellite communication systems. MF-TDMA is a hybrid of Frequency Division Multiple Access (FDMA) and Time Division Multiple Access (TDMA) technologies: an MF-TDMA *frame* consists of a set of adjacent-frequency carriers (with appropriate guard bands separating them); each carrier is divided into fixed-length time slots using TDMA. Thus, a *channel* in MF-TDMA is identified by a time-frequency slot in the frame. Each slot can carry a single fixed-size data packet called a *cell*. The slots in the MF-TDMA frame are shared by the UTs, under the control of the MAC scheduler: each UT may be assigned one or more slots to transmit its traffic (depending on its capacity needs), with the restriction that a UT cannot transmit on multiple carriers simultaneously, in order to reduce the complexity and cost of the transmitter. Thus, each UT's maximum transmission rate is limited to the carrier rate (the capacity of a single carrier).

Figure 2 shows the structure of a typical MF-TDMA frame consisting of signaling slots and payload slots. The signaling slots carry capacity request messages (see Section 2.1.4) and are also used by the UTs to synchronize their transmit times (clocks) to the frame. The payload slots carry the user traffic. Figure 2 shows a possible set of slot assignments for 18 UTs; the number in each slot identifies the UT authorized to transmit in that slot. For simplicity, the slots assigned to a UT

are usually contiguous, as shown in the figure.

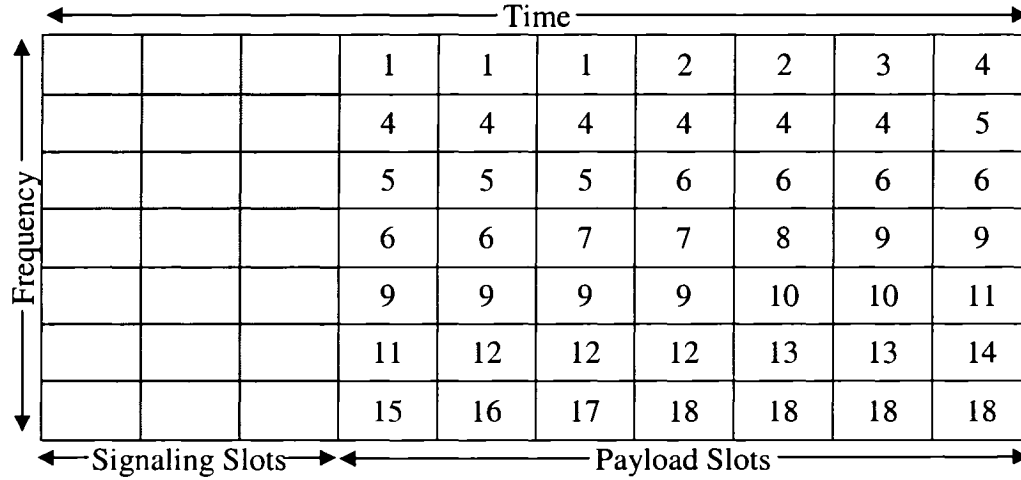


Figure 2: Typical MF-TDMA frame structure with slot assignments

MF-TDMA provides better trunking efficiency than simple FDMA or TDMA due to greater flexibility in assigning the available channels (slots), which translates into better (more economical) use of the available bandwidth. To demonstrate this point, consider for example an MF-TDMA frame with N time slots and M carriers; the number of available channels (slots) is thus NM . To provide the same number of channels, an FDMA access scheme would require NM carriers instead of M carriers, which is wasteful of the frequency spectrum. Similarly, a TDMA access scheme would require NM time slots in the TDMA frame instead of N time slots; assuming that the same frame duration as in MF-TDMA is maintained (in order to maintain the same slot transmission rate expressed in slots per second), the transmission rate on the TDMA carrier would have to be M times greater than on the MF-TDMA carriers because by definition, TDMA transmits all NM slots in the frame using a single carrier. This implies an increased UT antenna size and/or transmission power in order to support the higher transmission rate, thereby increasing the UT's cost.

2.1.3. Multi-beam satellite architectures and inter-beam switching

The main stumbling block for the widespread deployment of BSA networks in the

past has been the high cost of the UT's outdoor unit, consisting of the satellite dish and transceiver. The problem stems from the fact that a single satellite beam was used to cover a large geographic area; low-directivity satellite antennas were used to disperse the transmitted signal over this area. The UT's antenna therefore had to be large (and consequently expensive) in order to receive this weak signal from the satellite. The UT also required a powerful and expensive transmitter because the gain of the satellite antennas was low due to their low directivity. To circumvent these problems, the single-beam satellite architecture was replaced by a multi-beam architecture relying on highly directive multi-beam antennas [8]. Instead of providing coverage of a geographic area using a single, wide beam, the new architecture uses multiple narrow beams to provide continuous coverage of the same area. The reference BSA network architecture shown in Figure 1, for example, has M beams.

The multi-beam architecture has three main advantages. First, since the beams are relatively narrow, the signal transmitted by the satellite is not widely dispersed, reducing the required UT dish size. Second, high-directivity antennas generally have high gains, reducing the UT's transmission power requirements. Third, multi-beam technology allows frequency reuse between beams, leading to increased network capacity for given satellite physical resources (frequency spectrum). The increased capacity in turn allows a larger number of users to be supported, thus driving down costs by the principle of economies of scale. Note that a one-to-one correspondence between uplink beams and downlink beams is not necessary; for instance, a geographic area served by a single downlink beam may be served by multiple uplink beams (because downlink beams are typically wider than uplink beams).

The advantages of multi-beam technology come at the price of increased system complexity, because the satellite must establish an uplink cell's destination to determine on which downlink beam to forward it. An obvious solution is to simply forward the cell on all the downlink beams; this is not desirable, however, because it is wasteful of bandwidth. For optimal bandwidth use, the cell should only be transmitted on the downlink beam serving the destination UT or gateway;

the satellite must therefore perform on-board inter-beam switching. The form of this switching depends on the architecture of the satellite payload: non-regenerative vs. regenerative.

A non-regenerative payload is alternatively referred to as a transparent payload or repeater. Such a payload is reduced to the RF communication section (transponder). Each uplink carrier is simply frequency-translated and amplified into a downlink carrier without any demodulation or reconditioning (regeneration). Transparent payloads lead to the well-known “bent-pipe” satellite architecture. Pseudo-bent-pipe architectures can be obtained by providing frequency switching at the physical layer (see Section 2.1.3.1); such architectures allow limited connectivity between beams.

With a regenerative payload, the uplink carriers are demodulated to baseband and remodulated for the downlink transmission. Such an on-board processing helps to improve the transmission performance (by reducing the bit error rate) and offers full flexibility regarding the inter-beam connectivity supported by the satellite (see Section 2.1.3.2).

2.1.3.1. Non-regenerative payload: frequency switching

Since non-regenerative payloads do not demodulate the uplink carriers, the satellite does not have access to the cell headers so it cannot use the cell destination information embedded in these headers to perform switching. At most, non-regenerative payloads can translate uplink carriers into downlink carriers in different downlink beams, a process known as frequency switching. The upstream uplink MF-TDMA frame is split into two or more sub-frames, each consisting of a group of consecutive uplink carriers; ideally there should be as many sub-frames as there are downlink beams in the network in order to support full inter-beam connectivity. The satellite directs all the carriers in each sub-frame to a different downlink beam. The frequency-switching matrix used by the satellite may be dynamic, i.e., the mapping of uplink carriers to downlink beams may change periodically in response to shifting load patterns in the BSA network, with the constraint that the total number of uplink carriers directed to any

downlink beam must not exceed the downlink beam's transponder bandwidth. The frequency switching concept was initially adopted as an intermediate solution on the road towards regenerative payloads with full on-board switching (OBS) capabilities. It was hoped that frequency switching would be simpler to implement than full OBS but this has not proven to be the case because frequency switching introduces considerable complexity into the upstream uplink MAC scheduling process. The UT must keep track of the current mapping of uplink MF-TDMA carriers to downlink beams and must transmit cells only on the carriers corresponding to the desired destination beams; if the UT does not have any slot assignments on the carriers corresponding to the desired destination beam, that destination becomes unreachable. Furthermore, the UT's slot assignments for different downlink beams (and for the same downlink beam) must not overlap in time (recall from Section 2.1.2 that a UT cannot transmit on multiple carriers simultaneously). This non-overlap requirement also applies in single-beam systems but it does not pose a significant problem in this case because each UT is assigned a single contiguous block of slots (as noted in Section 2.1.2), so the non-overlap requirement can be satisfied simply by ensuring that the number of slots assigned to each UT does not exceed the total number of slots in a carrier.

2.1.3.2. Regenerative payload: on-board switching

Regenerative payloads demodulate the uplink carriers and recover the baseband data, enabling the satellite to access the cell headers (including the cell's destination downlink beam) and use them for inter-beam switching, much like a conventional packet switch. Such OBS offers significant advantages over frequency switching. First and foremost, it decouples the switching process from the upstream uplink MAC process, significantly simplifying the latter (see Section 2.1.3.1). Second, it allows a finer switching granularity to be achieved because individual cells are forwarded independently (cell-level granularity), as opposed to frequency switching, which forwards entire carriers (carrier-level granularity). The only drawback of OBS is that a regenerative payload is required, which is

more complex than a non-regenerative payload; the current industry trend is clearly towards regenerative payloads, however, so this issue is becoming less and less relevant as the technology matures.

2.1.4. Dynamic capacity allocation and scheduling

In Section 2.1.2, we described the MF-TDMA multiple access scheme used to share the capacity of an uplink beam among all the UTs in the beam. We noted that the UTs are assigned time-frequency slots in the MF-TDMA frame to transmit their traffic, but we did not consider the mechanics of this slot assignment process in detail. We now describe the process.

One option is to make each UT's slot assignment static (fixed). This method is simple and provides good link utility with applications generating traffic with constant or nearly constant rates (e.g., audio/video streaming). Unfortunately, most multimedia applications, such as web browsing, file transfers and videoconferencing, generate variable or even bursty traffic. In this case, the fixed assignment scheme would lead to either low utility but good QoS performance, if the assigned capacity is close to the application's peak transmission rate, or high utility but inadequate QoS performance due to queue buildups (increased delays) and/or overflows (packet losses), if the assigned capacity is close to the application's mean rate. Neither of these outcomes is acceptable, therefore the slot assignment process should dynamically adapt to the input traffic variations in order to ensure high resource utility while maintaining adequate QoS performance for the user traffic. This dynamic adaptation is achieved as follows. Each UT periodically requests transmission capacity in the form of MF-TDMA slots from a (typically centralized) *scheduler*; the request amounts reflect the UT's capacity needs, which depend on the current input traffic conditions. The capacity request message is transmitted either in the signaling area of the MF-TDMA frame (see Section 2.1.2), or may be "piggybacked" onto a payload slot if the UT already has a slot assignment in the current frame. The scheduler periodically runs a capacity assignment (scheduling) algorithm to distribute the time-frequency slots in the frame among the UTs based on their requests; the scheduler broadcasts the

resulting slot assignment map to the UTs in a capacity assignment message identifying the slot(s) assigned to each UT. This request-assignment process is called Dynamic Capacity Allocation (DCA), and constitutes the heart of the research presented in this thesis. Since all UTs in the same uplink beam must compete for access to the frame slots, a UT may not obtain all its requested capacity because the scheduler must also consider requests from the other UTs.

Due to the satellite propagation delay and request processing time at the scheduler, there is a lag between the time when a request is sent and when the corresponding capacity assignment is received; as noted in Section 1.2, this lag is called the DCA cycle time or request-to-assignment time. It can potentially reduce DCA performance because it slows the DCA scheme's responsiveness to changing input traffic conditions at the UTs: any changes to the capacity request amounts are only reflected in the capacity assignments received one DCA cycle time later. The request processing time at the scheduler is typically negligible compared to the satellite propagation delay, hence the DCA cycle time is determined mostly by the location of the scheduler: ideally, it should be located on board the satellite, in which case the cycle time is equal to one satellite hop delay (recall from Section 2.1.1 that this is the time required for a ground-satellite-ground hop, approximately 270 ms for the geosynchronous-orbit satellites used for BSA). An on-board scheduler adds complexity and computational and power requirements to the payload in order to implement the scheduling algorithm, however; this can only be justified in conjunction with other on-board processing functions such as inter-beam switching (see Section 2.1.3.2). There is a trend towards satellites with on-board processing capabilities so it is expected that the scheduler will eventually migrate to the satellite; currently, however, it is typically located on the ground (in the MCS), where computational and power resources are more readily available. In this case, the DCA cycle time doubles to two satellite hop delays (one for the capacity request and another for the corresponding assignment).

As in the case of the upstream uplink described above, the downstream downlink may also benefit from the use of DCA: consider the situation in which traffic from

multiple SPs connected to different gateways is destined for UTs located in the same downlink beam. In this case, a bottleneck may occur at the satellite because there may be insufficient capacity on the downlink beam to forward the traffic from all the gateways. This situation can be handled in one of two ways. First, if the satellite has a regenerative payload with an on-board queuing capability, temporary bottlenecks are handled by queuing the packets from the gateways until there is sufficient downstream downlink capacity available to forward the packets to the UTs. Second, if the satellite cannot perform on-board queuing, bottlenecks at the satellite must be avoided altogether by ensuring that there is sufficient downstream downlink capacity available in the destination downlink beam *before* allowing the gateways to transmit their traffic to the satellite. Thus, the gateways must use DCA to compete for the available downstream downlink capacity, just as the UTs compete for the upstream uplink capacity. DCA for the downstream downlink is beneficial even with on-board queuing because, as in any queuing system, it is possible for the queues to overflow. While in this thesis we focus on a DCA scheme for the upstream uplink, the DCA objective for the downstream downlink is essentially the same (to efficiently share limited transmission resources among multiple traffic sources), therefore the DCA scheme proposed in Chapter 3 could be modified to operate on the downstream downlink as well.

There may be a single scheduler for the entire BSA network (a centralized approach), or a separate scheduler for each SP (a distributed approach). In the latter case, each scheduler is only responsible for assigning capacity to the UTs “belonging” to a given SP; these UTs form the SP’s *domain*. Since the UTs in an uplink beam may belong to different SPs, the MF-TDMA frame for the uplink beam is divided into sub-frames controlled by the various schedulers: each SP’s scheduler is only permitted to assign time-frequency slots within its own sub-frame. The relative sizes of the sub-frames, and thus, the total upstream uplink capacity available for each SP’s UTs, are statically determined by the service agreements in place between the SPs and the BSA network operator. The relative sizes of the sub-frames may also change in a quasi-dynamic fashion, e.g., in response to changing traffic patterns throughout the day. The distributed

scheduling approach requires considerable coordination and introduces signaling complexity into the system, due to the multiple schedulers and the division of the MF-TDMA frame into sub-frames. To reduce the system complexity and consolidate the control of the overall scheduling process, the centralized approach is preferred, in which a single scheduler (typically located in the MCS) serves all the UTs in the system. Since this scheduler controls the entire MF-TDMA frame, dividing it into sub-frames is no longer required, which allows for more flexibility in the treatment of capacity requests from UTs belonging to different SPs. For instance, a certain SP's UTs may receive preferential treatment of their capacity requests, depending on the service agreement in place between the SP and the BSA network operator. Due to the advantages described above, the centralized scheduling approach is assumed in this thesis.

2.2. IP QoS architectures

In this section we describe the concept of end-to-end QoS in IP networks and the current state of the main IP QoS architectures used to provide end-to-end QoS: Integrated Services (IntServ) and Differentiated Services (DiffServ). We give emphasis to the DiffServ architecture due to its current dominance as the IP QoS architecture of choice.

2.2.1. End-to-end QoS

IP QoS requirements may be expressed in terms of packet delay, delay variation (jitter) and/or loss constraints, peak and/or average transmission rates or any combination thereof. End-to-end IP QoS is defined as the QoS experienced between the two endpoints of an IP connection. IP connections routinely span multiple networks; for instance, during a voice-over-IP (VoIP) session, the traffic is typically routed through the local user's access network / edge ISP network (e.g., America Online, Videotron), one or more core/backbone networks (e.g., UUNET/WorldCom/MCI, AT&T, Sprint), and finally the remote user's edge ISP network. To provide end-to-end QoS to the connection, each network in the

traffic path must guarantee a certain level of QoS, e.g., an upper bound on the delay experienced by traffic packets between entering and exiting the network. If even a single network in the path fails to provide the specified QoS level, the end-to-end QoS cannot be maintained. The typical steps involved in setting up an IP connection requiring QoS are:

- i. The initiating party (source) starts an application session (e.g., VoIP) with the destination party and negotiates the end-to-end QoS parameters for the session.
- ii. The session layer communicates the end-to-end QoS requirements of the application to the connection/transport layer.
- iii. The connection/transport layer uses various QoS signaling protocols (cited in Sections 2.2.2 and 2.2.3 below) to initiate a communication among the networks to be involved in the new IP connection. These networks verify the availability of the required resources (by executing a call admission control procedure) to meet the application's end-to-end QoS requirements. This process is governed by the Service Level Agreements (SLAs) [9] in place among the various network operators, and between the edge ISPs and their subscribers (end users); these SLAs define in detail the services provided by the network operators to each other and to the end users. An important element of an SLA is the Traffic Conditioning Agreement (TCA) [10], which specifies acceptable traffic profiles and performance metrics for the user traffic, the actions performed for non-compliant traffic (e.g., packet dropping) and any traffic shaping services provided by the network operator.
- iv. If each network operator involved in the new IP connection agrees to provide the required QoS, the appropriate resources are reserved in all the networks; the connection is thus admitted and the application session can proceed. If, however, one or more of the networks is/are unable to fulfill the QoS requirements (either because the SLA/TCA is violated or because there are insufficient network resources available), the connection cannot be set up and is rejected.

The most important point to note from the above discussion is that end-to-end QoS provisioning requires cooperation (communication) between multiple networks and between multiple protocol layers within the same network. More details on the tasks to be performed in various networks/layers can be found in [11]. The scope of the research presented in this thesis is limited to QoS-related processes that take place at the MAC layer in the BSA network segment of an end-to-end IP connection (i.e., DCA). This means that no further consideration will be given to processes taking place at the other layers mentioned above. Moreover, QoS support in the other (terrestrial) network segments spanned by the IP connection has been extensively researched elsewhere, notably in many Internet Engineering Task Force (IETF) Request for Comments (RFC) documents.

2.2.2. Integrated Services (IntServ)

The IntServ architecture [2] was the first IP QoS architecture proposed. It complements traditional best-effort service by using a signaling scheme to set up an end-to-end path (connection) for each individual traffic flow. The Resource Reservation Protocol (RSVP) [12] [13] [14] is the protocol of choice for IntServ signaling. It is used in conjunction with call admission control to reserve forwarding resources (buffer space and bandwidth) in each network node (router) along the end-to-end connection's path without affecting the performance of any already-established connections. Reservations are made on a per-flow basis so each router must generate and maintain per-flow state information.

In addition to the end-to-end path reservation, the IntServ architecture proposes two classes of service defining the treatment of packets at network nodes: the Guaranteed service and the Controlled-Load service.

- The Guaranteed service [15] places a guaranteed upper bound on the end-to-end delay of TCA-compliant traffic regardless of the network load (i.e., the QoS is always maintained). Compliant traffic is never discarded due to queue overflows provided that the traffic's parameters remain within the ranges specified in the TCA. Note that the upper delay bound is defined by the

network, not the user application; it is up to the application to cope with this bound or to cancel the connection if it is not acceptable.

- The Controlled-Load service [16] offers no quantitative QoS guarantees but makes a qualitative assurance that the QoS performance will not be much worse than the QoS performance on an unloaded network. The QoS is thus guaranteed on a statistical basis (forwarding resources are allocated based on the average traffic load as measured in the recent past).

The IntServ architecture relies on a number of mechanisms (such as packet classifiers, rate controllers and schedulers) that must be implemented in all network routers, which must also support the RSVP signaling protocol, provide call admission control functionality and maintain per-flow state information (as stated previously). The complexity, scalability and deployment-related problems of IntServ have caused this architecture to fall out of favor recently.

2.2.3. Differentiated Services (DiffServ)

The DiffServ architecture [3] [4] was developed in response to the complexity, scalability and deployment-related concerns regarding IntServ. Unlike IntServ, DiffServ does not rely on end-to-end per-connection resource reservations, making it more scalable in large networks because the network nodes (routers) no longer need to maintain per-connection state information and forwarding resources (buffer space and bandwidth). Instead, the DiffServ model requires that each network node provide well-defined packet forwarding treatments called per-hop behaviors (PHBs) to broad classes of aggregate traffic flows based on the DiffServ Code Point (DSCP) field embedded in the IP packet header. Though DiffServ is not itself an end-to-end QoS architecture, it is nevertheless possible to provide end-to-end QoS by implementing the DiffServ PHBs in all the network nodes. With DiffServ, the inter-network signaling required to provide end-to-end QoS (see Section 2.2.1) may rely on the new Session Initiation Protocol (SIP) [17].

Based on the TCA (see Section 2.2.1), each traffic packet entering a DiffServ domain is classified into one of the DiffServ Classes of Service (CoS) and marked

with the corresponding DSCP. The traffic is then conditioned to ensure compliance with the TCA; as a result of this conditioning, packets may be remarked to a different (usually inferior) CoS, delayed for traffic shaping purposes, or simply dropped. Only the boundary nodes in a DiffServ domain perform these traffic classification and conditioning functions; the internal nodes simply forward the packets according to the requirements of the PHB specified by each packet's DSCP. For this reason, it is said that DiffServ pushes the computational complexity to the network boundary, aiding scalability.

We provide a brief summary of the currently defined PHBs below:

- Expedited Forwarding (EF) PHB [18]: This PHB specifies that the aggregate departure rate of packets from a DiffServ node must equal or exceed a configurable rate independently of any other traffic transiting the node. EF traffic may temporarily preempt other traffic if necessary to meet this requirement. The EF PHB is typically used in conjunction with CAC and traffic conditioning to provide a low-loss, low-latency and low-jitter service.
- Assured Forwarding (AF) PHB Group [4] [19]: This PHB group actually defines four AF classes: AF1, AF2, AF3 and AF4; traffic in each class is forwarded independently and each class receives a configurable minimum amount of forwarding resources (buffer space and bandwidth). In addition, within each AF class, three levels of drop precedence are defined; in case of congestion, the packets with higher drop precedence are dropped first. The AF PHB group is intended to deliver TCA-compliant traffic with a very low loss probability; noncompliant traffic may also be delivered, but with a higher loss probability or an increased delay. Delay and jitter constraints are not specified for the AF PHB group; the EF PHB should be used instead for delay and jitter-sensitive traffic.
- Default (DE) PHB: This PHB is equivalent to conventional best-effort service and does not offer QoS assurances of any kind.

2.3. DiffServ-capable scheduler and UT architectures

Current BSA network architectures are not designed to provide DiffServ QoS support. The major difficulty in providing DiffServ support in a BSA network is that the access of the UTs to the upstream uplink is controlled by satellite MAC protocols (i.e., DCA) so the link capacity seen by each UT is variable and depends on the UT's current capacity assignment (see Section 2.1.4). This makes meeting the provisioning requirements of the DiffServ PHBs more difficult than in terrestrial networks, where the link capacity at the output of a network node (e.g., router) is fixed. In a BSA network, the DCA and the UT's traffic queuing strategy must therefore be carefully considered in order to meet the DiffServ provisioning requirements. In this section we describe our proposed architectures for the scheduler and the UT, the two key components involved in DCA and in providing DiffServ support on the upstream uplink of the BSA network. The design philosophy behind our proposed architectures is that the BSA network becomes a DiffServ domain; the ingress nodes into this domain are the UTs on the return link (upstream direction) and the gateways on the forward link (downstream direction).

2.3.1. Scheduler

As described in Section 2.1.4, the role of the scheduler is to manage the upstream uplink capacity (i.e., to assign MF-TDMA frame slots to the UTs) in order to ensure efficient upstream uplink utilization while providing adequate QoS to the user traffic. For the reasons discussed in Section 2.1.4, centralized (as opposed to distributed) scheduling is assumed. Even with centralized scheduling, however, in a multi-beam environment each uplink beam conceptually has its own scheduler that serves only the UTs in that beam. This is because each uplink beam's transmission resources (carriers) are independent of other beams' resources (frequency reuse is one of the main advantages of multi-beam architectures). In reality, these conceptually separate schedulers are physically co-located in a single unit and collectively labeled "scheduler". In the following

discussions, we describe the conceptual scheduler for a single uplink beam, with the understanding that there are multiple such schedulers in a multi-beam environment.

Due to the current popularity and open nature of the DVB-RCS return link specification [6] (see Section 1.1), it is assumed that the scheduler's operation is based on the DVB-RCS capacity request categories, summarized below:

- Continuous Rate Assignment (CRA): The scheduler uses this category to assign a fixed number of slots to the UT in each MF-TDMA frame for as long as the UT requires this capacity. The number of slots assigned per frame may be determined by an initial CRA request from the UT (during connection establishment) or may be statically defined in the SLA. In the case of an initial CRA request, the UT does not need to renew it for subsequent frames: the request remains in effect until the UT explicitly cancels it.
- Rate Based Dynamic Capacity (RBDC): Like CRA, this capacity is expressed in slots per frame. An RBDC assignment for a given UT remains in effect for a certain number of frames (the validity period) and is then reset to zero in the absence of a new RBDC request from the UT. Each RBDC request overrides any previous RBDC request from the same UT. The UT is guaranteed to receive the requested RBDC up to a maximum number of slots per frame (the Maximum RBDC). If the UT's RBDC request exceeds the Maximum RBDC configured for that UT, the scheduler treats the excess portion of the request as a VBDC request (see below). The main difference between CRA and RBDC is that the CRA amount may be configured statically in the SLA, making the CRA capacity available to the UT in each frame regardless of whether the UT actually has traffic to transmit. RBDC, on the other hand, is assigned only in response to an explicit RBDC request from the UT; thus, acquiring RBDC capacity entails an initial delay (the DCA cycle time described in Section 2.1.4) before the capacity is assigned. For delay-sensitive traffic, this initial delay may be unacceptable, in which case CRA should be used. The drawback of using CRA is that it may lead to an inefficient utilization of the MF-TDMA frame capacity because the CRA

capacity is assigned regardless of whether the UT actually has traffic to transmit; RBDC, on the other hand, allows for statistical multiplexing. For this reason, BSA network operators typically charge the user more for CRA than for RBDC.

- Volume Based Dynamic Capacity (VBDC): This capacity is expressed in slots, not slots per frame. Unlike RBDC requests, VBDC requests are cumulative: the scheduler adds each new request amount from a UT to a cumulative counter of pending VBDC requests from that UT. Any VBDC amount assigned to the UT is subtracted from this cumulative request counter; hence, the cumulative request amount may be assigned to the UT gradually over a number of frames depending on the availability of frame capacity. Unlike for CRA and RBDC, the scheduler does not guarantee the availability of VBDC; the scheduler first satisfies the CRA and RBDC requests from all the UTs and only subsequently assigns VBDC if there are still unassigned frame slots available.
- Absolute Volume Based Dynamic Capacity (AVBDC): This capacity can be considered as a special case of VBDC. It is similar to VBDC except that AVBDC requests are not cumulative: each new request overrides the previous one from the same UT. For simplicity, we use the VBDC designation to refer to both capacity types.
- Free Capacity Assignment (FCA): This capacity is not requested by the UTs. If there are any free slots left in the MF-TDMA frame after the CRA, RBDC and VBDC requests from all the UTs have been satisfied, the scheduler distributes the remaining free slots among the UTs.

From the above description, it is evident that the scheduler does not treat all capacity requests equally. The assignment priority order is, from highest to lowest, CRA and RBDC \rightarrow VBDC \rightarrow FCA. The purpose of having multiple capacity request categories is to increase the scheduler's flexibility in dealing with traffic having various QoS requirements. Furthermore, the standardization of these capacity request categories (in the DVB-RCS standard) makes the scheduler's design independent of the type of network layer in use (e.g., IP,

ATM). All that is required for the scheduler to support a given network layer is an appropriate *mapping* between the network layer's QoS architecture and the DVB-RCS capacity request categories. In this thesis, an IP-based network layer using the DiffServ QoS architecture is assumed; the required mapping in this case is between the DiffServ PHBs and the DVB-RCS capacity categories. This mapping is described in more detail in Section 2.3.2.

Since the CRA and RBDC capacity categories offer capacity guarantees as described above, the sum of all the guaranteed capacity (for all the UTs) must not exceed the MF-TDMA frame capacity:

$$\sum_{all\ UTs} (CRA + \text{Maximum RBDC}) \leq \text{Total Uplink Frame Capacity}$$

The scheduler maintains a UT data list for all the UTs that are currently logged into the BSA network. The list entry for each UT contains state information used in the scheduling process:

- A UT ID that is unique among all the UTs in the same uplink beam.
- The currently configured CRA (in slots/frame).
- The currently configured Maximum RBDC (in slots/frame).
- The current RBDC request amount (in slots/frame).
- The validity period of the UT's RBDC requests (i.e., the number of frames until the requests expire).
- The expiration timer for the current RBDC request (in frames). It is decremented in each frame and reset to the validity period when a new RBDC request is received from the UT. If the expiration timer reaches zero, the current RBDC request expires (i.e., the request amount is zeroed).
- The cumulative VBDC request amount (in slots).

The scheduling process uses two algorithms. The first handles UT data list updates (as a result of incoming capacity requests or changes to the configured CRA or Maximum RBDC). The second algorithm handles the actual capacity assignment in conformance with the characteristics of the DVB-RCS capacity request categories defined above. This capacity assignment algorithm involves

three passes through the UT data list. On the first pass, CRA and RBDC (the guaranteed capacity types) are assigned; on the second pass, VBDC is assigned; finally, on the third pass any remaining capacity is distributed as FCA. Both the UT data list update and the capacity assignment algorithms are invoked periodically, e.g., at the start of each frame (the scheduling may also be performed per multiple frames). Since the capacity requests from the UTs may arrive at any time during the frame, the scheduler stores these requests in an incoming request queue and processes them at the beginning of the next frame. Once the scheduling process is complete and all the slots in the MF-TDMA frame have been assigned, the assignment information is broadcasted to the UTs in an allocation map (MAP) message [5]. To reduce the size of this MAP message (and thus, the amount of signaling overhead), the slots assigned to each UT form a contiguous block in the MF-TDMA frame and the capacity assignment for each UT consists simply of a scalar indicating the total number of slots assigned (not separate assignments for each DVB-RCS capacity category) and the time-frequency coordinates of the first slot in the block.

Figure 3 shows a block diagram of the scheduler with all the elements/structures described above.

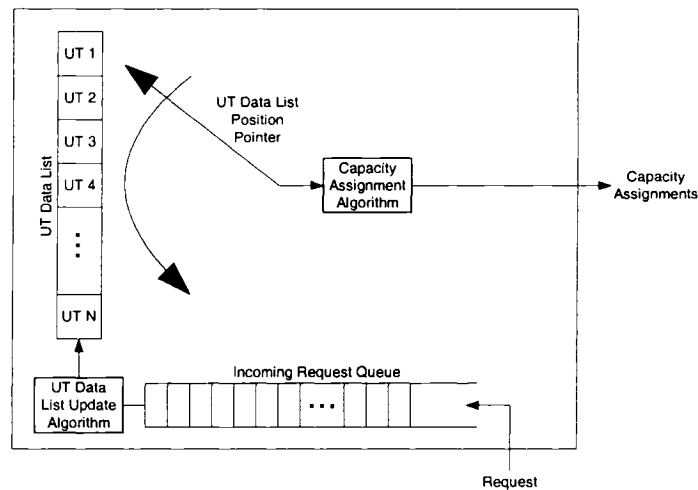


Figure 3: Scheduler block diagram

2.3.2. User terminal

In our discussion of the scheduler in Section 2.3.1, we made no mention of DiffServ (except to state that a mapping is required between the DiffServ PHBs and the DVB-RCS capacity categories). This is because the UT, not the scheduler, is responsible for implementing DiffServ QoS mechanisms. Giving this responsibility to the UT is a logical approach because the UT has the most up-to-date information concerning the current input traffic conditions and is thus best equipped to make QoS-related decisions. This approach also simplifies the scheduler's architecture, which is desirable, especially if the scheduler is located on board the satellite where computing and power resources are limited (see Section 2.1.4). Furthermore, as described in Section 2.3.1, it makes the scheduler's design independent of the type of network layer (e.g., IP, ATM) and QoS architecture in use.

Figure 4 shows a block diagram of the UT's traffic path, the DCA and QoS-related structures and the interactions between them. Solid lines represent traffic flows and dotted lines represent signaling. We present the functions of the various structures below.

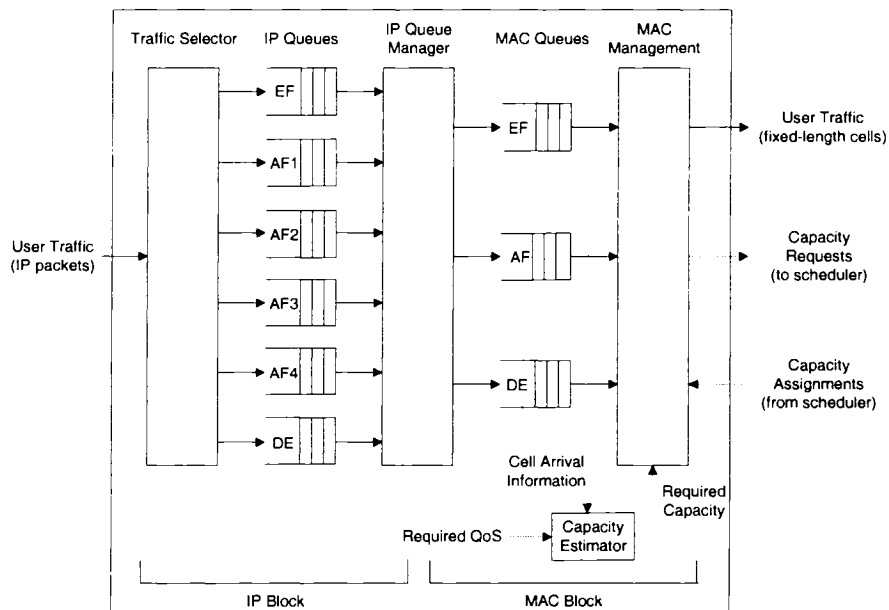


Figure 4: UT traffic path, DiffServ and DCA structures

- Traffic Selector: This module performs DiffServ traffic classification and conditioning functions, which are necessary because the UT is an ingress node into the DiffServ domain formed by the BSA network. The traffic selector accepts user IP packets and classifies them into DiffServ Classes of Service (EF, AF1, AF2, AF3, AF4 or DE) based on rules specified in the TCA negotiated with the user (for instance, the packet's source IP address and/or port number). Following classification, each packet is marked with the DSCP corresponding to the chosen CoS. Alternatively, the classification and marking may be performed inside the user's own network, in which case the traffic selector does not need to perform these functions. Next, the traffic in each CoS is metered to determine whether it conforms to the TCA. If so, the traffic is inserted into the appropriate IP queue (there is one IP queue for each CoS). If, however, the traffic is noncompliant, it may be demoted to another CoS (e.g., DE), shaped, or simply dropped to enforce TCA compliance [10]. If the user wishes to transmit traffic beyond the current TCA, the user could renegotiate the TCA to accommodate this traffic; for instance, the UT could obtain higher values for CRA or Maximum RBDC (see Section 2.3.1) for a higher cost. The possibility of TCA renegotiation depends on the local policy and the availability of adequate signaling protocols in the user's network.
- IP Queues: The IP queues hold the packets arriving from the Traffic Selector. There is one queue for each DiffServ CoS.
- IP Queue Manager: The IP queue manager segments packets from the IP queues into fixed-length cells in a format suitable for transmission over the upstream uplink, and inserts these cells into the corresponding MAC queues. The cell size coincides with the MF-TDMA slot size. For the EF and DE classes, segmentation is a straightforward process because there is a one-to-one correspondence between the IP and MAC queues involved. The process is slightly more complicated for the four AF classes, however, because there is only a single AF MAC queue in order to simplify the capacity request calculation and capacity distribution (see MAC Management module below). Consequently, a queuing discipline (e.g., WFQ) is used to schedule packets

from the AF1, AF2, AF3 and AF4 IP queues into a single output prior to segmentation.

- MAC Queues: The MAC queues hold MAC cells until they are transmitted on the upstream uplink.
- MAC Management: This module is at the heart of the UT's DCA process. It is responsible for calculating capacity request amounts, transmitting the requests, receiving the capacity assignments from the scheduler, and distributing the assigned capacity among the three MAC queues. The request calculation and capacity distribution algorithms are detailed in Chapter 3.
- Capacity Estimator: This module uses information regarding the cell arrivals in the MAC queues and the required QoS for each MAC queue in order to estimate the UT's capacity needs to achieve the required QoS. Further details are given below.

Figure 5 shows a detailed diagram of the Capacity Estimator module. It takes as inputs the number of cell arrivals per frame and required QoS for each MAC queue (specified in terms of the QoS criterion defined in Section 3.1). Based on the cell arrival information, the Traffic Parameter Estimator sub-module estimates various parameters of the input traffic in each queue (e.g., the mean arrival rate).

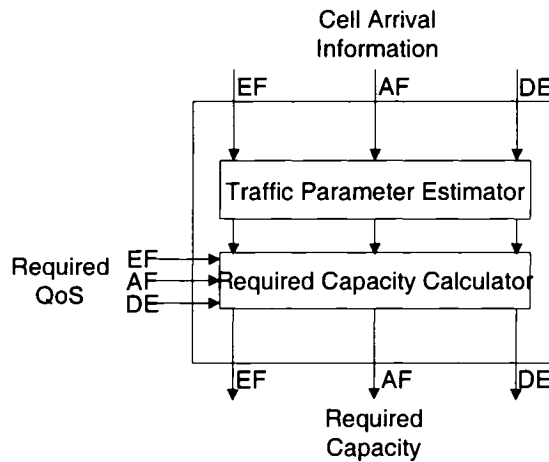


Figure 5: UT Capacity Estimator module

The estimated traffic parameters are fed into the Required Capacity Calculator sub-module, which uses them in conjunction with the required QoS for each MAC

queue in order to calculate the service rate (capacity) needed for each queue to achieve the required QoS. This calculation is based on an analytical model of the queuing performance. The required capacity values for all three MAC queues are passed to the MAC Management module, which uses them for the calculation of the capacity requests and the distribution of the capacity assigned by the scheduler among the MAC queues. All the algorithms and equations used in the MAC Management and Capacity Estimator modules (for the traffic parameter estimation, required capacity calculation, and capacity request and distribution) are described in detail in Chapter 3.

The UT's two queuing levels (IP and MAC) promote a logical partitioning of the UT structure into functional blocks, as shown in Figure 4. The IP block consists of the Traffic Selector, IP queues and IP Queue Manager; this block is responsible for implementing DiffServ traffic classification, conditioning and queuing mechanisms. Since DiffServ is an IP-based architecture, these mechanisms are logically implemented at the IP level. The IP Queue Manager provides the interface between the IP and MAC blocks. The MAC block consists of the MAC queues, MAC Management module and Capacity Estimator; it is responsible for the implementation of the UT's DCA functions, i.e., traffic parameter estimation, required capacity calculation, and capacity request and distribution (as noted above).

Recall from Section 2.3.1 that since the capacity requests follow the format specified by the DVB-RCS standard, a mapping is required between the DiffServ PHBs and the DVB-RCS capacity categories. More specifically, the request amount calculated for each UT MAC queue (EF, AF and DE) must be mapped to at least one DVB-RCS category. In order to devise an effective mapping, the requirements of the DiffServ PHBs must be matched to the characteristics of these categories, as described below.

- The EF PHB requires a bandwidth guarantee and is typically used to provide a low-loss, low-latency and low-jitter service. EF traffic tends to be smooth. CRA is a good match for these requirements. RBDC may be used to supplement the CRA, but not completely replace it because the scheduling lag

involved in obtaining an RBDC assignment (see Section 2.3.1) conflicts with the EF PHB's low-latency requirement. The reason for using RBDC to supplement CRA is that RBDC capacity usually costs less; users may choose to accept the delay involved in obtaining an RBDC assignment in exchange for lower-cost service.

- The AF PHB group also requires a bandwidth guarantee but there are no delay and jitter constraints. AF traffic may be bursty. RBDC is a good match for these requirements: it offers a bandwidth guarantee (up to the Maximum RBDC) and can efficiently handle bursty traffic because RBDC is assigned dynamically in response to explicit requests from the UTs. VBDC may be used to supplement the RBDC, but since VBDC does not offer a bandwidth guarantee, there is once again a tradeoff between the service quality and cost.
- The DE PHB has no requirements of any kind. It should not be mapped exclusively to FCA, however, in order to avoid completely starving the DE traffic when there is no FCA available. Due to this possibility of starvation, the DVB-RCS standard specifies that no traffic class should be mapped exclusively to FCA. Consequently, DE traffic should be mapped to a combination of VBDC and FCA.

Table 1 summarizes the mappings described above.

	CRA	RBDC	VBDC	FCA
EF	X	X		
AF		X	X	
DE			X	X

Table 1: Possible mappings between DiffServ PHBs and DVB-RCS capacity categories

Chapter 3 : A DYNAMIC CAPACITY ALLOCATION SCHEME FOR DiffServ QoS SUPPORT

In this chapter, we present our proposed DCA scheme, which aims to provide high satellite bandwidth utility and satisfy user QoS requirements, while maintaining reduced signaling overhead and algorithmic complexity. In Section 3.1, we define an appropriate QoS performance measure that can be used to derive the basic QoS parameters of packet delay and loss. In Section 3.2, we provide a survey/summary of previous work on DCA schemes for satellite networks. In Section 3.3, we present our modeling of the upstream uplink input traffic in the BSA network, justifying the selected model's validity using results from the literature and assumptions regarding the typical operational environment of the UTs. In Section 3.4, we describe the moving-average estimator of the mean traffic arrival rate, a key DCA input. In Section 3.5, we describe the network overload situation and its expected negative performance impact, followed by our derivation of the network overload probability and mean overload time. Finally, in Section 3.6 we present our DCA scheme. A key component of the scheme is an analytical expression for the capacity required by each UT to achieve a target QoS for the user traffic; this expression is derived using an analytical queuing model based on the input traffic model previously selected in Section 3.3.

3.1. QoS performance measure

Recall from Section 2.3.2 that each UT has three MAC queues for the EF, AF and DE DiffServ service classes. In order to design a DCA scheme that allows some QoS requirement (criterion) to be satisfied for each queue, it is necessary to define an appropriate QoS performance measure to quantify the queuing performance, and thus, to determine whether the criterion is satisfied. We could simply use the basic QoS parameters of packet queuing delay and loss probability as our performance measures; alternatively, we could define a single unified measure from which these basic parameters are easily derived. Selecting the latter approach, we define the QoS measure as the queue size survival function, i.e., the

probability of the queue length Q (in packets) exceeding a value n , $\Pr\{Q>n\}$, for an infinite queue. This QoS measure is convenient due to its close relationship with the basic QoS parameters: $\Pr\{Q>n\}$ can approximately represent the packet loss probability of a finite queue of length n , since a packet arriving at a full queue must be discarded; furthermore, for a queue service rate C and length Q , the average queuing delay is approximately $D = Q/C$, so $\Pr\{Q>n\}$ is equivalent to $\Pr\{D>n/C\}$, the delay survival function.

The QoS criterion expressed in terms of a delay requirement is that the queuing delay D must not exceed some value $d=n/C$, or equivalently, that the queue length Q must not exceed n , otherwise an *outage* (QoS violation) is said to occur. The outage probability must be maintained below a certain value. The QoS criterion for each of the UT's MAC queues can therefore be stated as follows:

$$\Pr\{Q>n_k\}_k \leq P_{OUT} \quad (1)$$

where k represents the queue type (1, 2 or 3 for the EF, AF and DE queue, respectively) and P_{OUT} is the desired outage probability, i.e., the probability of the queue length exceeding n_k . In other words, Equation (1) simply states that, with a probability of $1 - P_{OUT}$, queue k does not exceed n_k packets. Queues requiring lower delays (sizes), e.g., the EF queue, have lower n_k values. A queue is deemed to receive satisfactory QoS if the above criterion is satisfied.

3.2. Background and previous work

Since the scheduler assigns MF-TDMA frame slots to the UTs in response to their capacity requests, the slot assignment is demand-based; furthermore, any free (unassigned) slots remaining in the frame after all the UT requests have been satisfied are distributed among the UTs in a process called Free Capacity Assignment (FCA). This hybrid assignment scheme, consisting of both a demand-based component and FCA, is called Combined Free/Demand Assignment Multiple Access (CF-DAMA) [20], developed in order to improve the delay and link utility performance of a geosynchronous satellite link supporting multiple users with bursty input traffic. Since the CF-DAMA concept does not specify the actual algorithms used for the requests and assignments, the

CF-DAMA family encompasses a wide range of possible DCA schemes, including the one proposed in this thesis.

An alternative to the structured demand-assignment model of CF-DAMA is random slot access, in which the UTs simply transmit their traffic in randomly selected slots, risking collisions with other transmitting UTs. As there is no need to wait for capacity assignments, this method offers immediate slot access, at the cost of poor satellite link utility. In the case of the Slotted Aloha (S-Aloha) random access scheme, for example, the maximum theoretical link utility is 36.8% [21]; this is considered unacceptable for most satellite networks, given the scarcity and high cost of satellite bandwidth. Consequently, DCA schemes based on CF-DAMA have enjoyed increased popularity. Since demand-based assignment is combined with FCA, slot access is immediate at lower network loads (since the bulk of the frame slots are then assigned using FCA), thus matching the latency performance of S-Aloha (at similar loads); as the load increases, CF-DAMA's demand-based component gains more importance, allowing the link utility to approach 100%.

The original CF-DAMA design in [20] has since been subjected to detailed performance analysis, and has enjoyed a number of enhancements and refinements, resulting in numerous variants. These variants differ mainly with regard to the request signaling strategy. In [22], the authors provide an analysis of the average delay and delay variation performance of a CFDAMA-PB scheme with capacity requests piggybacked (PB) onto payload slots, as described in Section 2.1.4. This request signaling strategy benefits the UTs already having slot assignments, since these UTs can immediately transmit capacity requests, while other UTs must wait to receive slot assignments via FCA in order to transmit their requests. Piggybacking the requests onto payload slots results in low signaling overhead, since the requests need not be carried separately in the MF-TDMA frame's signaling area (see Section 2.1.2). In [23] and [24], two other request signaling strategies are proposed and compared by simulation and analysis, both strategies relying on the MF-TDMA frame's signaling area. The first pre-assigns (reserves) a signaling slot for each UT in the network; since the number of UTs

typically exceeds the number of signaling slots in the frame, each UT can only access its pre-assigned (PA) slot every n frames. This request signaling strategy is called CFDAMA-PA. The second strategy allows random access (RA) to the signaling slots, risking collisions between capacity requests from multiple UTs; this strategy is called CFDAMA-RA. It is shown that the PA strategy offers excellent delay-throughput performance for a low/medium UT population size, while the RA strategy becomes more efficient as the population size increases. Considering the advantages of the PB strategy, reported in [22] and further analyzed in [25], the authors recommend the use of a hybrid PB/RA strategy capable of providing optimal performance across a wide range of population sizes. This recommendation has been considered in the DVB-RCS standard, which supports both PB and RA strategies, in addition to the PA strategy (for full flexibility).

The hybrid PB/RA strategy described above is further analyzed in [26] under the name “combined request” (CFDAMA-CR). Two methods are proposed to control access to the RA slots, in order to prevent their overuse by certain UTs to the detriment of others, and thus, to improve the overall delay-throughput performance. Since controlling access to the RA slots can be considered unfair, a new request signaling strategy is proposed in [27], based on round-robin (RR) assignment of the signaling slots to the UTs; this is a variant of the PA strategy. The performance of the new strategy is assessed by simulation for various input traffic models (Poisson, Pareto ON-OFF, exponential ON-OFF) and compared with the performance of the other request signaling strategies described above. While performance for Poisson traffic is invariant with regard to the signaling strategy, improved delay-throughput performance is obtained with the RR scheme for the ON-OFF traffic models; the distribution of ON and OFF state sojourn times (Pareto or exponential) was found to be less relevant.

In [28], a predictive CF-DAMA scheme is proposed (PRDAMA). The scheme uses the variation in the number of packets in each UT’s input queue at successive sampling instants in order to predict the traffic’s future arrival trend; this method is based on the local linear approximation (LLA) method [29]. The prediction is

used to perform FCA: the free capacity is assigned to the UTs in proportion to each UT's predicted future traffic arrival trend. Simulation results illustrate the effectiveness of PRDAMA, with beneficial impact on the average packet queuing delay and jitter performance.

The literature cited above is concerned primarily with request signaling strategies to improve the performance of CF-DAMA schemes; the actual determination of the capacity request amounts is not considered in great detail, however. Typically, the request amount is simply set to the number of packets currently in the UT queue minus the outstanding request amount; this is true even for the more advanced PRDAMA scheme [28], since the scheme's predictive component is only used for FCA. In addition, a fundamental limitation of the above CF-DAMA schemes is that they are all based on a single queue per UT; such schemes cannot be used in a BSA environment supporting service differentiation for multimedia applications, since service differentiation requires managing capacity among multiple queues. The present thesis addresses this issue in the context of the DiffServ framework (Section 2.2.3), and suggests specific enhancements to CF-DAMA in terms of the UT architecture/functionality and the algorithms and equations used to determine the capacity request and assignment amounts, in accordance with the research issues defined in Section 1.3. In particular, the main advantages of our proposed DCA scheme with regard to those reviewed in the above literature survey are summarized below:

- UT architecture reflecting the DiffServ service classes (see Section 2.3.2).
- Built-in support for DiffServ service classes in the capacity request and assignment algorithms and equations, as described in Section 3.6 below.
- Novel capacity request and assignment equations taking the required QoS into account: an analytical model of the queuing performance is used to calculate the amount of capacity required in order to satisfy the QoS criterion of Equation (1) (Section 3.1) for each of the UT's MAC queues, as described in Section 3.6.1 below.
- Effectiveness with both SRD and LRD input traffic types, i.e., the proposed DCA scheme is not tied to a particular traffic type. This is not true of the

PRDAMA scheme described above, for example, because the performance of its predictive component depends on specific characteristics of the input traffic (in particular, strong time correlation).

- Built-in mechanisms for optimizing the DCA performance in view of the various tradeoffs described in Section 1.3; in particular, these mechanisms include the moving-average sliding window size described in Section 3.4 below, and the request transmission threshold described in Section 3.6.3.

3.3. Input traffic modeling

Although UT hardware and service costs have been steadily decreasing due to advances in multi-beam antenna technology (see Section 2.1.3), the costs are still not low enough for BSA technology to be widely adopted by individual users, especially in geographical areas where terrestrial access alternatives (e.g., cable, DSL) are available. Consequently, we assume that the UTs are typically deployed in multi-user environments so the equipment and service costs are shared, lowering the per-user cost; the traffic entering a UT thus represents an aggregate of IP users. We assume that each UT is connected to several small IP-based LANs as shown in Figure 6; each small LAN consists of one or more user devices (e.g., PCs) connected to the UT through an aggregation device such as a low-cost Internet access router (available commercially from several manufacturers). Each LAN may represent, for example, a home network in an apartment building (given the increasing number of multiple-computer households) or a corporate network in an office building.

Due to the nature of LAN technology, LAN traffic is characterized by an alternation of bursts and idle periods. During the bursts, one of the stations (user devices) connected to the LAN transmits its data at a constant rate, the LAN rate (e.g., 10, 100 or 1000 Mbps for Ethernet); during the idle periods, none of the stations is transmitting. Consequently, the output traffic from each of the small LANs connected to the UT can be modeled by an ON-OFF source with packets generated at the LAN rate in the ON state, and no packets generated in the OFF state, as shown in Figure 6. The state sojourn times of the ON-OFF sources are

specified by a probability distribution. This distribution may be exponential, with probability density function (PDF) and mean given by Equation (2):

$$f(x) = \lambda e^{-\lambda x} \quad x > 0 \quad (2)$$

$$E(x) = 1/\lambda$$

In this case, the resulting traffic is said to exhibit short-range dependence (SRD). Alternatively, the distribution may exhibit the *heavy-tailed* property, as for example the Pareto distribution with PDF and mean given by Equation (3):

$$f(x) = \frac{ca^c}{x^{c+1}} \quad a \leq x < \infty \quad (3)$$

$$E(x) = \frac{ca}{c-1} \quad c > 1$$

The tail of the PDF decays more slowly, as suggested by the “heavy-tailed” terminology, and consequently, the variance is usually much higher; the Pareto distribution of Equation (3), for instance, has an infinite variance (but finite mean) for $1 < c < 2$. A heavy-tailed state sojourn time distribution results in traffic exhibiting long-range dependence (LRD) [30].

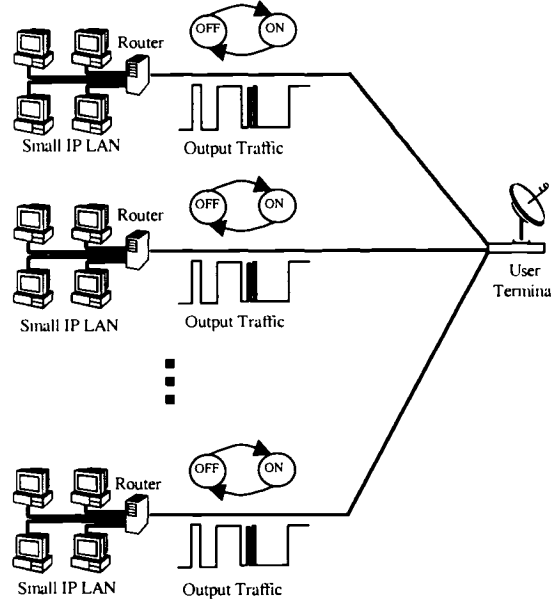


Figure 6: Typical UT LAN connectivity in a multi-user environment

The SRD and LRD properties refer to the relative decay rates of the traffic’s time autocorrelation function: the decay rate follows a power law for LRD traffic, and an exponential law (with a faster decay rate) for SRD traffic. The LRD property

indicates that the underlying traffic exhibits self-similarity, i.e., that the traffic appears the same when observed at many different time scales ranging from milliseconds to hours. LRD and self-similarity have significant implications for queuing performance: LRD input traffic in queuing systems generally results in longer queues (and thus, increased packet queuing delays and losses) than SRD traffic [30]. Measurements performed on real-world LAN traffic flows have shown that these flows exhibit LRD and thus, self-similarity [31] [32] [33]. Flow aggregates, however, may exhibit SRD behavior, depending on the number of aggregates. In this thesis, both SRD and LRD input traffic types are therefore considered.

Since the output traffic from each small LAN can be modeled as an ON-OFF source as described above, it follows that the aggregate traffic entering the UT from all the small LANs is a superposition of ON-OFF sources. Such a superposition can be modeled by a switched Poisson process, as described in [34] [35] [36]. In this thesis, we consider a two-state switched Poisson process as illustrated in Figure 7. The packet arrival processes in both states are Poisson processes with mean arrival rates λ_H and λ_L , so the packet inter-arrival times are exponentially distributed with means $1/\lambda_H$ and $1/\lambda_L$, respectively. The state with the higher λ value, λ_H , is called the “high” state; the other state is called the “low” state. The mean sojourn times in the high and low states are $1/r_H$ and $1/r_L$, respectively, where r_H and r_L are the mean transition (switching) rates from the high state to the low state and from the low state to the high state, respectively. The sojourn times are exponentially or Pareto-distributed, depending on the distribution of the ON and OFF times of the individual ON-OFF traffic sources. The resulting switched Poisson process is called a Markov-Modulated Poisson Process (MMPP) [37] in the former case and a Pareto-Modulated Poisson Process (PMPP) [38] in the latter case. If the ON and OFF times of the individual sources are exponentially distributed (as for SRD traffic), their superposition can be modeled by an MMPP [34] [35]; if the ON and OFF times are Pareto-distributed (as for LRD traffic), the superposition can be modeled by a PMPP [36].

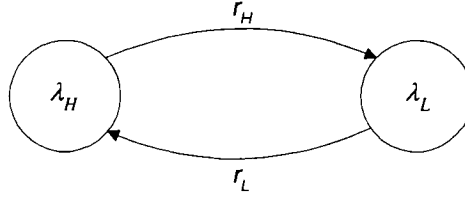


Figure 7: Two-state switched Poisson process model

In summary, the aggregate traffic at the input of the UT can be modeled by an MMPP (for SRD traffic) or PMPP (for LRD traffic). Consequently, we model the aggregate input traffic in each of the UT's three MAC queues by a two-state MMPP or PMPP traffic source. Each source switches (toggles) between its two states independently of the other sources, i.e., the state switching points are not synchronized. Each UT queue and its associated traffic source together form an MMPP/G/1 or PMPP/G/1 queuing system. The mean sojourn times, $1/r_H$ and $1/r_L$, are set greater than the DCA cycle time defined in Section 2.1.4, so that each traffic source can be assumed to remain in the same state for the entire duration of a DCA cycle. Consequently, the traffic within the DCA cycle behaves as simple (non-switched) Poisson traffic, so that within this time interval, the queuing system can be treated as an M/G/1 system, which lends itself more easily to analysis than an MMPP/G/1 or PMPP/G/1 system. This simplification is used in the derivation presented in Section 3.6.1 of the service rate (capacity) required for each of the UT's MAC queues in order to satisfy the QoS criterion of Equation (1) (Section 3.1).

3.4. Moving-average estimation of the mean traffic arrival rate

Recall from Section 2.3.2 that one of the functions of the UT's Capacity Estimator module (more specifically, the Traffic Parameter Estimator sub-module) is to estimate various parameters of the input traffic in each of the UT's MAC queues. Since the traffic can be treated as simple (non-switched) Poisson traffic during a DCA cycle, as described in Section 3.3, the only relevant traffic parameter is the mean arrival rate during the cycle, which is either λ_H or λ_L depending on the input

traffic source's current state. In this section, we describe a method for estimating the mean arrival rate in each queue (service class) using a moving time-average, also called a sliding-window average, of cell arrivals.

First, we define the following variables:

- L : The sliding window size in frames, i.e., the number of consecutive MF-TDMA frames over which the time average is taken.
- a_k : A list of length L storing a history of the number of cell arrivals in the UT's queue k for the last L frames. Element j of a_k ($j = 1, \dots, L$) is denoted by a_{kj} and represents the number of cell arrivals $j-1$ frames ago, with $j=1$ denoting the current frame (at the head of the list) and $j=L$, the oldest (at the tail). The list is updated once per frame: a_{kL} (the number of cell arrivals $L-1$ frames ago) is removed from the tail and the number of cell arrivals in the current frame is added to the head; this is the sliding-window mechanism. The moving time-average is then recalculated using the updated list.
- T_F : The MF-TDMA frame time, i.e., the frame's duration. We assume $T_F=26.5$ ms, typical for real-world satellite communication systems.

With the above definitions, the estimated mean traffic arrival rate in queue k , $\bar{\lambda}_k$, is given by the moving average:

$$\bar{\lambda}_k = \frac{\sum_{j=1}^L a_{kj}}{LT_F} \quad (4)$$

With a smaller window size L , the moving average covers a shorter time span (fewer frames) and thus, follows the instantaneous traffic arrival rate more closely, responding more quickly to traffic transients and to state changes of the switched Poisson input traffic source. This implies high variation in the estimated mean. As L is increased, the estimated mean becomes smoother, i.e., the estimate is of better quality, possibly leading to improved DCA performance; the downside is that the estimate responds more slowly to traffic variations such as the traffic source's state switching point, with the opposite effect of reducing DCA performance. As noted in Section 1.3.2, there is thus a tradeoff between the estimate's quality and its responsiveness; this tradeoff is investigated in Chapter 4

through the optimization of the window size in order to maximize DCA performance.

3.5. Network overload probability and mean overload time

As described in Section 3.3, the input traffic in the BSA network is modeled by two-state switched Poisson traffic sources (one source for each UT MAC queue). With the parameters of these sources (λ_H , λ_L , r_H and r_L) defined in Section 3.3, the mean rate λ of the traffic generated by each source is given by

$$\lambda = \frac{\frac{\lambda_H}{r_H} + \frac{\lambda_L}{r_L}}{\frac{1}{r_H} + \frac{1}{r_L}} \quad (5)$$

It follows that the aggregate mean traffic arrival rate in an uplink beam (from all the UTs in the beam) is simply given by multiplying the per-source mean rate of Equation (5) by the total number of traffic sources, which is equal to three times the number of UTs in the beam since each UT has three MAC queues with one source per queue.

If enough of the switched Poisson traffic sources in an uplink beam are simultaneously in their “high” state, the aggregate traffic arrival rate in the beam may momentarily exceed the beam’s total uplink transmission capacity, although the aggregate *mean* arrival rate does not. This situation is called a *network overload*, and persists until some of the traffic sources in the “high” state switch to the “low” state. During network overloads, the traffic arrival rates to the UT queues exceed the queue service rates because there is insufficient transmission capacity available in the affected uplink beam; the queues therefore become unstable (build up), resulting in increased queuing delays and/or packet losses (i.e., worse QoS performance). Consequently, the overloads’ occurrence frequency and duration both affect the performance; overloads should only occur rarely (i.e., the *overload probability* should be low) and their duration should be short, in order to minimize their performance impact. The overload duration is related to the state sojourn times of the traffic sources: longer sojourn times lead

to longer overload times, because more time elapses before traffic sources in the “high” state switch to the “low” state, thus ending the overload. As described in Section 3.3, LRD PMPP traffic, with its heavy-tailed Pareto sojourn time distribution, generally exhibits longer sojourn times with higher probabilities than SRD MMPP traffic (for the same mean sojourn time). The network overload times are thus presumably longer with PMPP input traffic (and LRD traffic in general); to determine the difference, the overload times with MMPP and PMPP traffic must be compared. The reduced QoS performance resulting from the prolonged overload times with PMPP traffic can be counteracted by lowering the network traffic load; this is investigated in Chapter 4. In this section, we derive expressions for the network overload probability and mean overload time; numerical results obtained using these expressions are presented in Chapter 4, along with corresponding simulation results for comparison and discussion.

3.5.1. Network overload probability

First, we define the following variables:

- N : The total number of UTs in the uplink beam.
- M^H : The total number of switched Poisson traffic sources in the uplink beam that are currently in the “high” state. The number of sources in the “low” state is given by $3N - M^H$.
- S : The total uplink transmission capacity (in cells/sec), obtained by dividing the number of payload slots in the MF-TDMA frame by the frame time (duration) T_F .

The aggregate traffic arrival rate is $M^H \lambda_H + (3N - M^H) \lambda_L$. It follows that the network overload probability can be succinctly expressed as

$$P\left\{M^H \lambda_H + (3N - M^H) \lambda_L > S\right\}$$

or equivalently,

$$P\left\{M^H > \frac{S - 3N\lambda_L}{\lambda_H - \lambda_L}\right\} \quad (6)$$

Equation (6) states that the desired probability is given by the survival function of

M^H . Since M^H is the number of switched Poisson traffic sources that are currently in the “high” state (out of a total $3N$ traffic sources), and the sources operate independently of each other (i.e., a source’s state changes do not affect the state changes of other sources), M^H is binomially distributed. Denoting the mean state sojourn time in the “high” and “low” state by $s_H = 1/r_H$ and $s_L = 1/r_L$, respectively, $\Pr\{M^H\}$ is thus the binomial probability mass function (PMF):

$$\Pr\{M^H\} = \binom{3N}{M^H} \cdot \left(\frac{s_H}{s_H + s_L} \right)^{M^H} \cdot \left(\frac{s_L}{s_H + s_L} \right)^{3N-M^H} \quad (7)$$

Consequently, Equation (6) is the binomial survival function, i.e., the one’s complement of the binomial cumulative distribution function (CDF). Since there is no simple closed-form expression for this CDF, the desired survival function can be calculated using the PMF of Equation (7):

$$P\left\{M^H > \frac{S - 3N\lambda_L}{\lambda_H - \lambda_L}\right\} = \sum_{M^H = \left\lfloor \frac{S - 3N\lambda_L}{\lambda_H - \lambda_L} + 1 \right\rfloor}^{3N} \binom{3N}{M^H} \cdot \left(\frac{s_H}{s_H + s_L} \right)^{M^H} \cdot \left(\frac{s_L}{s_H + s_L} \right)^{3N-M^H} \quad (8)$$

where $\lfloor \cdot \rfloor$ denotes the floor operator.

3.5.2. Mean network overload time

In this section, we derive an expression for the mean duration of a network overload, i.e., the mean overload time. Our derivation uses Markov analysis and is thus only valid for MMPP input traffic. The corresponding derivation for PMPP traffic is considerably more complex and is not considered in the thesis; instead, simulation is used to evaluate the mean overload time for PMPP traffic.

Let $M^H(t)$ be a continuous-time, integer-valued random process representing the number of traffic sources in the uplink beam that are in the high state at time t . $M^H(t)$ remains at a given value (state) until a state transition occurs in one of the traffic sources: if a source moves from the high to the low state, $M^H(t)$ decreases by 1 and conversely, if a source moves from the low to the high state, $M^H(t)$ increases by 1. Consequently, the sojourn time of $M^H(t)$ at a given value (state) is the minimum of the state sojourn times of the $3N$ traffic sources in the uplink beam. By definition, the state sojourn time of an MMPP source is exponentially

distributed so the state sojourn time of $M^H(t)$ is the minimum of $3N$ independent exponentially distributed random variables, which is also an exponential random variable¹. Consequently, since the state sojourn time of $M^H(t)$ is exponentially distributed, $M^H(t)$ is a Markov chain.

To determine the mean transition rate of $M^H(t)$ from state 0 to state 1, we recognize that such a transition signifies that one of the $3N$ traffic sources in the uplink beam has moved from the low state to the high state, with all the other sources in the “low” state. Since r_H and r_L are the mean transition rates of the traffic sources out of the high and low states, respectively, the mean transition rate of $M^H(t)$ from state 0 to state 1 is thus $3Nr_L$ (the minimum of $3N$ i.i.d. exponential random variables with rate r_L is also exponential with rate $3Nr_L$). Similarly, a transition from state 1 to state 2 signifies that one of the $3N - 1$ traffic sources that were still in the low state has just moved to the high state; consequently, the mean transition rate from state 1 to state 2 is $(3N - 1)r_L$. Generally, the transition rate from state m to state $m + 1$ is $(3N - m)r_L$. By the same reasoning, it is not difficult to see that the transition rate from state m to state $m - 1$ is mr_H . Consequently, since the only possible transitions out of state m are to the adjacent states $m+1$ or $m-1$ (i.e., $M^H(t)$ is a birth-and-death process), the mean transition rate out of state m is $(3N - m)r_L + mr_H$. The mean sojourn time in state m is thus $1 / [(3N - m)r_L + mr_H]$. Figure 8 shows the state transition diagram of $M^H(t)$.

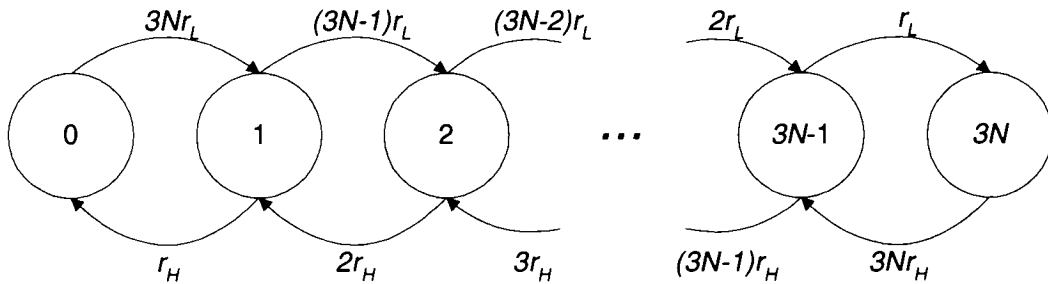


Figure 8: State transition diagram of $M^H(t)$

Next, from Equation (6), the threshold number of traffic sources that must be in the high state in order to cause a network overload is:

¹ From probability theory, the minimum of n independent exponential random variables with rates $\lambda_1, \dots, \lambda_n$ is an exponential random variable with rate $\lambda_1 + \dots + \lambda_n$.

$$M^{H*} = \left\lfloor \frac{S - 3N\lambda_L}{\lambda_H - \lambda_L} + 1 \right\rfloor$$

An overload occurs for $M^H(t) \geq M^{H*}$. We can thus group the states of $M^H(t)$ into two categories: the “overload” states for which $M^H(t) \geq M^{H*}$ and the “non-overload” states for which $M^H(t) < M^{H*}$. $M^H(t)$ can thus be abstracted as a two-state system, as shown in Figure 9:

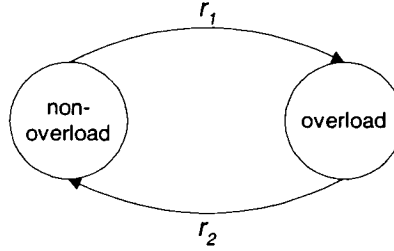


Figure 9: Two-state abstraction of $M^H(t)$

We are interested in calculating r_2 , the mean transition rate of $M^H(t)$ out of the overload state, because the mean overload time that we wish to derive is simply given by the inverse of this rate. We note that the only possible way for $M^H(t)$ to leave the overload state is by moving from the threshold state M^{H*} to state $M^{H*}-1$; this is because $M^H(t)$ is a birth-and-death process so only transitions between adjacent states are possible. r_2 is thus given by

$$\begin{aligned} r_2 &= \Pr\{M^H(t)=M^{H*} \mid M^H(t) \text{ is in overload}\} \cdot (\text{transition rate from state } M^{H*} \text{ to state } M^{H*}-1) \\ &= \Pr\{M^H(t)=M^{H*} \mid M^H(t) \geq M^{H*}\} \cdot (\text{transition rate from state } M^{H*} \text{ to state } M^{H*}-1) \end{aligned} \quad (9)$$

The first term in Equation (9) is the probability that $M^H(t)$ is in state M^{H*} given the knowledge that there is a network overload, i.e., that $M^H(t) \geq M^{H*}$. The second term in Equation (9) is the state transition rate from M^{H*} to $M^{H*}-1$. Thus, we have

$$r_2 = \frac{\Pr\{M^H(t) = M^{H*}\}}{\sum_{m=M^{H*}}^{3N} \Pr\{M^H(t) = m\}} \cdot M^{H*} r_H$$

The mean overload time is thus given by

$$\frac{1}{r_2} = \frac{\sum_{m=M^{H*}}^{3N} \Pr\{M^H(t) = m\}}{\Pr\{M^H(t) = M^{H*}\} \cdot M^{H*} r_H} \quad (10)$$

The state probabilities $\Pr\{M^H(t) = m\}$ in Equation (10) are binomially distributed,

as described in Section 3.5.1 (Equation (7)).

3.6. Proposed DCA scheme

As described in Section 2.3.2, each UT's DCA functionality is split between the Capacity Estimator and MAC Management modules; the former is responsible for estimating the input traffic parameters and using them to calculate the capacity required in order to satisfy the QoS criterion of Equation (1) (Section 3.1), and the latter is responsible for calculating the capacity request amounts, transmitting the requests to the scheduler, and distributing the capacity assigned by the scheduler among the UT's three MAC queues.

The scheduler uses the received capacity requests to assign the MF-TDMA frame slots to the UTs. Recall from Section 2.3.2 that if the scheduler's operation is assumed to conform to the DVB-RCS standard, each UT must perform a mapping between the capacity request amount for each MAC queue (EF, AF and DE) and the DVB-RCS capacity request categories. In this section, however, to avoid obscuring/diluting our discussion of the proposed DCA scheme, we exclude this mapping operation.

The work of Sections 3.1, 3.3 and 3.4 is now integrated coherently into our proposed DCA scheme. An evolutionary approach is taken in presenting the scheme using three *study cases*, each in itself a self-contained, operational DCA scheme. These three cases represent a logical evolution of the capacity request and assignment algorithms, allowing us to evaluate and compare multiple DCA design options that are in the end consolidated under the third (final) case. Thus, the first two cases are intended only as intermediate steps towards the final case. The three cases are described in the following subsections, and their anticipated performance differences are described. Subsequently, simulation is used in Chapter 4 to evaluate the performance of the three cases, and thus, to verify the anticipated differences; the simulation results show that the third (final) case represents the best compromise (tradeoff) between QoS performance, signaling overhead and algorithmic complexity.

3.6.1. Case 1 (C1): Frame-based requests with known traffic arrival rates (as baseline)

This case serves as our starting point, in order to establish a reference (baseline) performance level for subsequent comparison with the performance of the two other cases, whose design is derived from this baseline case.

Each UT sends a capacity request and receives an assignment from the scheduler every MF-TDMA frame; the DCA is thus performed on a frame-by-frame basis. The number of MF-TDMA slots requested by the UT for each of its MAC queues is equal to the number of new cell arrivals in the queue since the last request (in the previous frame); this strategy is similar to that used by the CF-DAMA schemes reviewed in the literature survey of Section 3.2. In addition to the number of new arrivals in each queue, however, our scheme's capacity request message also contains estimates of the capacity required for each queue in order to satisfy the QoS criterion of Equation (1) (Section 3.1). The scheduler uses these estimates to perform FCA of any remaining unassigned slots in the frame after all UT requests have been satisfied, as previously discussed in Section 3.2 (details of the scheduling algorithm are provided later in this section). As explained in Section 2.3.2, the estimation of the required capacity is performed by the UT's Capacity Estimator module, consisting of the Traffic Parameter Estimator and Required Capacity Calculator sub-modules. The operation of the former was detailed in Section 3.4. Below, we describe the operation of the latter. Recall from Section 3.3 that each UT MAC queue and its associated switched Poisson input traffic source together form an MMPP/G/1 or PMPP/G/1 queuing system that can be treated as a simpler M/G/1 system for the duration of a DCA cycle, because the mean state sojourn times of each traffic source are set greater than the DCA cycle time, so that each source can be assumed to remain in the same state during the DCA cycle. With a constant queue service rate, the M/G/1 system becomes M/D/1. Of course, in reality the service rate is *not* constant because the capacity assigned to each UT by the scheduler is variable, depending on the UT's capacity request amounts as well as the request amounts of the other UTs in the same uplink beam. For the moment, however, let us assume that the

service rate is constant. An upper bound for $\Pr\{Q \geq n\}$ for an M/D/1 system is given in [39]:

$$\Pr\{Q \geq n\} \leq e^{-\gamma n} \quad (11)$$

In the above expression, γ is the solution of the following equation:

$$\rho e^\gamma = \gamma + \rho \quad (12)$$

where $\rho = \lambda/C$ is the normalized queue load, λ is the mean packet arrival rate to the queue and C denotes the queue service rate. It is not difficult to derive an expression for C as a function of λ , n and $\Pr\{Q > n\}$. Noting that $\Pr\{Q > n\} = \Pr\{Q \geq n+1\}$ and using Equation (11) as an approximation rather than an upper bound, we obtain

$$\gamma \approx -\frac{\ln \Pr\{Q > n\}}{n+1} \quad (13)$$

Now, from Equation (12),

$$C = \lambda \frac{e^\gamma - 1}{\gamma} \quad (14)$$

Equations (13) and (14) allow us to calculate the capacity required for each of the UT's MAC queues in order to satisfy the QoS criterion of Equation (1) (Section 3.1): $\Pr\{Q > n\}$ in Equation (13) is set to the desired outage probability P_{OUT} , and n is set to the desired n_k value for queue k . Equation (14) indicates that the required capacity $C > \lambda$. Furthermore, as n is decreased, γ increases, so C also increases; intuitively, this is logical since lower n values indicate smaller required queue sizes and delays (in other words, more stringent QoS requirements).

The mean traffic arrival rate λ in Equation (14) can be estimated using a moving time-average of cell arrivals, as described in Section 3.4; this is applied to the second and third DCA cases, described in Sections 3.6.2 and 3.6.3 below. For this initial (baseline) case, however, we assume that the mean traffic arrival rates λ_H and λ_L of each MAC queue's input traffic source are known by the UT, and that the UT can detect the traffic source's state changes, allowing the required capacity C to be recalculated whenever a state change occurs, using Equation (14) with the λ value corresponding to the new state (λ_H or λ_L). The above assumptions (that the mean arrival rates and state switching points are known) are

admittedly somewhat unrealistic; their purpose is simply to provide a point of reference for assessing the DCA performance degradation, if any, resulting from estimating λ (rather than assuming it known) in our second and third DCA cases. As noted previously, the capacity assigned to each UT by the scheduler is variable; the UT may not always obtain all of its requested capacity due to contention from the other UTs in the same uplink beam, particularly during periods of high network load. Consequently, the service rate of each MAC queue fluctuates around the optimal value given by Equation (14); when the service rate is less than this optimal value, the QoS criterion of Equation (1) may be momentarily violated until the service rate increases. Nevertheless, Equations (13) and (14) provide a good guideline (approximation) of the capacity required. The method used by each UT to distribute the capacity assigned by the scheduler among its three MAC queues is described below, after defining the following variables:

- A_i : The number of MF-TDMA frame slots assigned to UT i by the scheduler in the current frame.
- D_{ik} : The number of slots distributed by UT i to its queue k (with $k = 1, 2, 3$ representing the EF, AF and DE queues, respectively).
- C_{ik} : The required capacity (service rate) calculated by UT i for its queue k using Equations (13) and (14).

The UT distributes the assignment A_i from the scheduler among its MAC queues in proportion to the required capacities C_{ik} :

$$D_{ik} = \left\lfloor \frac{C_{ik}}{\sum_{k=1}^3 C_{ik}} A_i \right\rfloor \quad (15)$$

Assuming that the queues are non-empty, Equation (15) is equivalent to Weighted Fair Queuing (WFQ) [40] with the weight of queue k equal to C_{ik} . The capacity distribution dynamically adapts to changing input traffic conditions as the C_{ik} 's are updated (recalculated) in response to state changes of the corresponding input traffic sources. Since queues with more stringent QoS requirements (lower n_k

values), e.g., the EF queue, have higher C_{ik} values, these queues receive a larger share of the assigned capacity A_i . Fractional slots cannot be distributed, so D_{ik} is rounded down to an integer number of slots, as indicated by the floor operator $\lfloor \cdot \rfloor$ in Equation (15); this rounding may result in a small number of leftover slots, since $A_i - \sum_{k=1}^3 D_{ik}$ may be > 0 ; in this case, the remaining slots are distributed using a priority queuing scheme [40] with the EF queue given the highest priority and the DE queue, the lowest (EF \rightarrow AF \rightarrow DE); this is reasonable, since the EF queue has the most stringent QoS requirements, followed by the AF queue (by definition, the DE queue has no QoS requirements and thus, receives the lowest priority).

Once per frame, the scheduler runs an algorithm to assign capacity to all the UTs in the uplink beam. The scheduler maintains a list of all these UTs, and performs three successive passes through the list, assigning capacity for the EF service class, followed by the AF class, and finally the DE class, as shown in the pseudo-code in Figure 10. We define the following variables:

- N : The total number of UTs in the uplink beam.
- R_{ik} : The capacity requested by UT i for its queue k (equal to the number of new cell arrivals in queue k since the last frame, as described previously).
- $R_{ik_CUMULATIVE}$: A cumulative counter of the outstanding (ungranted) capacity requested by UT i for its queue k . This counter is further described below.
- $carrier_capacity$: The total number of payload slots per carrier in the MF-TDMA frame. Recall from Section 2.1.2 that the UT cannot transmit on multiple carriers simultaneously in order to reduce the complexity and cost of its transmitter; the number of slots assigned to the UT per frame must therefore not exceed $carrier_capacity$.
- S_F : The total number of payload slots in the frame (the frame capacity).

```

 $S_F$  = total number of payload slots in frame
for  $i = 1, \dots, N$ 
     $A_i = 0$ 

for  $k = 1, \dots, 3$ 
    for  $i = 1, \dots, N$ 
         $R_{ik\_CUMULATIVE} = R_{ik\_CUMULATIVE} + R_{ik}$ 
         $assignment = \min(R_{ik\_CUMULATIVE}, carrier\_capacity, S_F)$ 
         $R_{ik\_CUMULATIVE} = R_{ik\_CUMULATIVE} - assignment$ 
         $S_F = S_F - assignment$ 
         $A_i = A_i + assignment$ 

```

Figure 10: Scheduling algorithm (pseudo-code)

We make the following observations regarding the above scheduling algorithm:

- The three-pass process ($k = 1, 2, 3$) gives higher assignment priority to service classes with lower k values: all the EF capacity requests ($k = 1$) are granted first, followed by the AF requests ($k = 2$), and finally the DE requests ($k = 3$). During periods of high network load, the scheduler may run out of frame slots to assign (S_F may be decremented to zero) before completing the third pass, in which case the service classes with higher k values are deprived of capacity.
- As noted above, the scheduler may run out of frame slots before completing the capacity assignment process. In order to avoid giving an unfair advantage to UTs nearer to the beginning of the UT list, the assignment process does not always begin with the first UT in the list ($i=1$) as indicated in Figure 10. Instead, the scheduler records the list position of the last UT served before running out of frame slots; the assignment process in the next frame begins with the next UT in the list. This behavior is not shown in Figure 10, in order to avoid cluttering the pseudo-code.
- The scheduler accumulates the capacity requests received from the UTs by maintaining cumulative request counters $R_{ik_CUMULATIVE}$ for all i and k . Each new capacity request R_{ik} is added to $R_{ik_CUMULATIVE}$; similarly, any capacity assigned to a UT is subtracted from the corresponding $R_{ik_CUMULATIVE}$, as shown in Figure 10. In simpler terms, the scheduler “remembers” capacity requests that have not yet been granted and eventually assigns this outstanding

capacity, possibly spread out over multiple frames (depending on the availability of capacity in the system and the assignment restriction imposed by the carrier capacity limit described above).

If there are unassigned (free) slots remaining in the frame after all the capacity requests have been granted (i.e., if $S_F > 0$ upon completion of the scheduling algorithm in Figure 10), the scheduler performs free capacity assignment (FCA). The free slots are assigned in proportion to each UT's required capacity values C_{ik} ; denoting the number of free slots by F , we have

$$A_i = A_i + \left\lfloor \frac{\sum_{k=1}^3 C_{ik}}{\sum_{i=1}^N \sum_{k=1}^3 C_{ik}} F \right\rfloor \quad (16)$$

As in Equation (15), the rounding in Equation (16) may result in a small number of leftover (unassigned) slots, which are distributed among the UTs in a round-robin fashion.

3.6.2. Case 2 (C2): Frame-based requests with estimated traffic arrival rates

This case is defined primarily in order to assess the DCA performance degradation, if any, resulting from estimating the mean traffic arrival rate rather than assuming it known as in C1; this comparison is performed in Chapter 4.

Recall that in the baseline DCA case C1 (Section 3.6.1), we assumed that the mean traffic arrival rates λ_H and λ_L of each MAC queue's input traffic source are known by the UT, and that the UT can detect the traffic source's state changes, allowing the required capacity C_{ik} to be recalculated whenever a state change occurs, using Equation (14) with the λ value corresponding to the new state (λ_H or λ_L). Now, for our second case C2, a more practical approach is used: we incorporate the work from Section 3.4 into the DCA scheme, allowing λ in Equation (14) to be *estimated* using the moving time-average of cell arrivals; the value of this average is updated every frame. Apart from estimating λ , C2 is otherwise identical to C1.

3.6.3. Case 3 (C3): Reduced request and assignment frequency (signaling) and simplified scheduling algorithm

This final case is defined with two objectives in mind: first, to reduce the required signaling overhead (as compared to the first two cases C1 and C2); and second, to reduce the complexity of the scheduling algorithm. Both of these objectives are desirable in practical BSA networks: reducing the signaling overhead makes satellite transmission bandwidth previously used for signaling available for user traffic, decreasing the service cost per unit bandwidth; similarly, reducing the complexity of the scheduling algorithm simplifies its implementation, reducing computational requirements and thus, cost. In Chapter 4, the performance of C3 is compared to that of the previous cases in order to determine whether the reduced signaling and complexity of C3 incur a performance penalty.

As noted above, the first goal of this DCA case is to implement a mechanism allowing the amount of signaling required for the capacity requests and assignments to be reduced. This is accomplished by reducing the request and assignment frequency; recall that in C1 and C2, the DCA is performed on a frame-by-frame basis, meaning that each UT sends a capacity request per frame. Now, in C3, a request for service class (queue) k is only sent if the required capacity for queue k in the current frame, $C_{ik_CURRENT}$, differs by more than a specified percentage from the required capacity at the time of the previous capacity request, $C_{ik_PREVIOUS}$ (recall that the required capacities are calculated using Equation (14)). Denoting the relative change in C_{ik} by Δ_{ik} ,

$$\Delta_{ik} = \frac{|C_{ik_CURRENT} - C_{ik_PREVIOUS}|}{C_{ik_PREVIOUS}} \quad (17)$$

Each UT i calculates Δ_{ik} for $k = 1, \dots, 3$ once per frame. The UT sends a capacity request for queue k only if $\Delta_{ik} > \Delta_{TH}$, where Δ_{TH} is the *request transmission threshold*. At the scheduler, a request remains valid until it is replaced by a new request, i.e., if in the current frame the scheduler does not receive a new request from UT i for queue k , the UT's request for queue k is set equal to the last request received for queue k from the UT.

As Δ_{TH} is increased, the amount of signaling decreases because fewer capacity

requests and assignments are transmitted per unit time. The inconvenience is that the scheduler must rely on older request information for the capacity assignment process; depending on the Δ_{TH} setting, this older information may no longer reflect the current input traffic conditions at the UT, resulting in sub-optimal capacity assignments (and thus, decreased performance). In other words, the DCA becomes less responsive to input traffic variations as Δ_{TH} is increased. As noted in Section 1.2, a tradeoff (compromise) is thus required between DCA performance and signaling overhead. Since Δ_{TH} allows the signaling overhead to be adjusted, it is the primary parameter used to investigate this tradeoff. The moving-average window size L described in Section 3.4 also affects the signaling overhead, albeit indirectly: as L is increased, the average traffic arrival rate $\bar{\lambda}_k$ becomes smoother (less variable), as does the required capacity C_{ik} due to its linear dependence on $\bar{\lambda}_k$ through Equation (14). Consequently, the relative changes Δ_{ik} in C_{ik} are generally smaller, meaning that Δ_{ik} is less likely to exceed a given Δ_{TH} , thus decreasing the request frequency and signaling overhead. L could therefore be used as a secondary parameter for investigating the performance-signaling tradeoff; Δ_{TH} is the preferred parameter, however, due to its more direct relationship to the signaling overhead, as described above. The optimization of Δ_{TH} and L is performed in Chapter 4.

The second goal of our final DCA case is to simplify the scheduling algorithm as compared to the algorithm used in C1 and C2, described in Section 3.6.1 (Figure 10). Simplification is especially important if the scheduler is located on board the satellite where computing and power resources are more limited, but is also desirable with a ground-based scheduler. The simplified algorithm assigns the entire frame capacity (slots) purely in proportion to each UT's required capacity values C_{ik} . Recall that in the previous cases (C1 and C2), only the *free* capacity was assigned in this manner using Equation (16); the bulk of the frame capacity was assigned using the scheduling algorithm in Figure 10. In our final case, the capacity A_i assigned to UT i is thus simply given by

$$A_i = \left\lfloor \frac{\sum_{k=1}^3 C_{ik}}{\sum_{i=1}^N \sum_{k=1}^3 C_{ik}} S_F \right\rfloor \quad (18)$$

where S_F denotes the total number of payload slots in the frame, as defined previously. The entire scheduling algorithm is thus elegantly reduced to Equation (18), greatly simplifying the scheduler's operation. As before, any leftover slots due to rounding in Equation (18) are assigned to the UTs in a round-robin fashion. In addition to providing simpler operation, the new scheduling algorithm also reduces the signaling overhead independently of the request transmission threshold mechanism described previously (Equation (17)). Recall that in our previous DCA cases (C1 and C2), each capacity request message contains:

- The originating UT's unique identifier i .
- The capacity request amounts R_{ik} for $k = 1, \dots, 3$, equal to the number of new cell arrivals in queue k since the previous request.
- The required capacity C_{ik} for $k = 1, \dots, 3$, calculated using Equation (14).

In C1 and C2, the R_{ik} 's were used by the scheduling algorithm in Figure 10. Since the simplified algorithm (Equation (18)) only uses the C_{ik} 's, the R_{ik} 's are no longer included in the capacity request messages, reducing their size, and thus, the signaling overhead. Furthermore, in Equation (18), only the *sum* of the C_{ik} 's over all k is used, so the UT can calculate and send this sum instead of the individual C_{ik} values in order to further reduce the signaling overhead (this is also true for C1 and C2). The reduced complexity of C3's scheduling algorithm may come at the price of decreased DCA performance; as described in Section 1.2, a performance-complexity tradeoff may therefore be required, and is investigated in Chapter 4.

Chapter 4 : PERFORMANCE EVALUATION

In this chapter, we use mainly simulation (but also some analytical) results to evaluate the performance of the DCA scheme proposed in Section 3.6, including the three defined cases. Simulation is used for the optimization of the DCA elements/parameters described in Section 1.3; in particular, this optimization includes the selection of a case from among the three defined cases (in view of the performance – algorithmic complexity tradeoff), and the setting of the traffic averaging window size (in view of the quality-responsiveness tradeoff defined in Section 3.4) and request transmission threshold (in view of the performance-signaling tradeoff). In Section 4.1, we describe the simulation model's configuration and select appropriate simulation scenarios allowing us to evaluate the desired aspects of the DCA performance, facilitating the aforementioned optimization. Next, in Section 4.2 we present and analyze the simulation results (comparing them to analytical results where appropriate), drawing conclusions regarding the DCA optimization. Finally, in Section 4.2.4 we consider the case of fast-switching input traffic (see Section 1.3.6), as opposed to the slow-switching traffic considered hitherto; simulation is used to determine how fast-switching traffic affects the DCA performance, and thus, whether the proposed DCA scheme must be adjusted to compensate.

4.1. Simulation model description

To evaluate the DCA performance, we developed a simulation model of the BSA network's upstream uplink traffic path and DCA-related functionality, using OPNET network simulation software. The simulation model includes a centralized scheduler and a number of UTs with switched Poisson input traffic sources generating the uplink traffic, as described in Section 3.3; the architectures and functionalities of the various components in the simulation model conform to the descriptions given in the previous chapters of the thesis. The performance results collected by the model include the average cell queuing delays in the UT MAC queues, the queue size survival functions, the network overload probability,

mean overload time and overload time survival function. Since upstream traffic enters the BSA network via the UTs and exits via the gateways, the total delay experienced by upstream traffic in the BSA network consists of the aforementioned UT queuing delay, the cell transmission time and the satellite hop delay. Since the cell transmission time and satellite propagation delay are fixed and known, our study focuses on the queuing delay.

4.1.1. Model configuration

To maintain reasonable simulation run times and computer memory requirements, we only simulate a single uplink beam. This does not limit the generality of our results, however, because in a multi-beam environment each uplink beam's transmission resources (carriers) are independent of other beams' resources, so the DCA process is simply duplicated: each uplink beam conceptually has its own scheduler that serves only the UTs in that beam, as described in Section 2.3.1. Also in the interest of maintaining reasonable run times and memory requirements, we simulate a scaled-down BSA network consisting of fewer UTs and a correspondingly smaller MF-TDMA frame than in typical real-world networks. Our results and conclusions, however, also apply to full-size networks. The configuration of our simulation model is presented below. This configuration is common to all the simulation scenarios described subsequently in Section 4.1.2.

User terminals: There are 32 UTs, of which 24 are "low-traffic" and the remaining 8 are "high-traffic". As these designations imply, the traffic sources in the high-traffic UTs generate traffic with a higher mean rate than the sources in the low-traffic UTs, as specified by our network loading strategy in Section 4.1.2.1. The reason for using two types of UTs instead of a single homogeneous group is to better reflect the conditions in real-world networks, where users often have different bandwidth requirements.

Satellite hop delay: The satellite hop delay is the propagation time required for a single ground-satellite-ground hop, 270ms for a geo-synchronous satellite. We assume that the scheduler is ground-based, so the process of making a capacity request and receiving an assignment requires two satellite hops (540ms); this is

the DCA cycle time or request-to-assignment time.

Input traffic sources: As described in Section 3.3, each UT's aggregate traffic flow (from the UT's users) is modeled by a switched Poisson process; we use two-state MMPP and PMPP traffic sources to model SRD and LRD traffic, respectively. There are three identical sources per UT, each generating input traffic for one of the UT's three MAC queues (EF, AF and DE). Recall from Section 3.5 that the mean rate λ of the traffic generated by each source is given by Equation (5), copied below for convenience:

$$\lambda = \frac{\frac{\lambda_H}{r_H} + \frac{\lambda_L}{r_L}}{\frac{1}{r_H} + \frac{1}{r_L}}$$

The values of λ_H and λ_L of each traffic source are chosen so as to obtain various total network loads (from all the traffic sources), as specified by our network loading strategy in Section 4.1.2.1. We set $\lambda_H = 5\lambda_L$, i.e., the mean traffic rate in the “high” state is five times that in the “low” state; this difference creates burstiness in the generated traffic. We use equal mean state sojourn times in both states: $1/r_H = 1/r_L = 3.25$ seconds, approximately 6 times the DCA cycle time specified above ($6 \times 0.540 = 3.24$ seconds), in accordance with our assumption in Section 3.3 that the mean state sojourn times are greater than the DCA cycle time.

MF-TDMA frame: The frame consists of 8 carriers, each with 32 payload time-slots, for a total of 256 payload slots per frame. The frame time (duration) T_F is 26.5 ms, which is typical for existing satellite networks. The slot size is 53 bytes, coinciding with the cell size in Asynchronous Transfer Mode (ATM) networks [40]; numerous existing satellite networks are ATM-based because ATM was the transport architecture of choice when these networks were designed. We therefore use this slot size for compatibility purposes, but we emphasize that the BSA architectures described in this thesis are based on IP and DiffServ, not ATM. With the frame parameters given above, the total upstream uplink information transmission rate² is 4 Mb/s, or 512 kb/s per carrier. As noted in Section 2.1.2, a

² The information transmission rate is the useable rate for user traffic, excluding coding, signaling and other transmission overhead.

UT cannot transmit on multiple carriers simultaneously in order to reduce the complexity and cost of its transmitter, so the UT's maximum transmission rate is 512 kb/s (the carrier capacity).

QoS parameters: Recall from Section 3.1 that the queue size n_k (in cells) is used in Equation (1) to specify the required QoS for each UT MAC queue (service class), with $k = 1, 2, 3$ denoting the EF, AF and DE queue, respectively. We set $n_1=75$, $n_2=115$ and $n_3=150$, with the outage probability $P_{OUT} = 10^{-3}$. Thus, the EF, AF and DE queues must not exceed 75, 115 and 150 cells, respectively, with a probability of $1-10^{-3} = 0.999$.

4.1.2. Simulation scenarios

We use the following simulation scenarios to evaluate and highlight various aspects of the DCA performance, in accordance with the research issues and DCA optimization defined in Section 1.3.

4.1.2.1. Network loading

To evaluate and compare the performance of the proposed DCA cases under various network loading conditions ranging from light to heavy, simulations are run for normalized network loads of 50, 60, 70, 75, 80, 85, 90 and 95% of the total uplink capacity (4 Mb/s, as given in Section 4.1.1). Performance differences between the DCA cases are expected to become more apparent as the load is increased, since the DCA is more heavily stressed. The load is split evenly between the low-traffic and high-traffic UTs: the 24 low-traffic UTs generate 50% of the network load and the remaining 8 high-traffic UTs generate the other 50%. The load generated by a single low-traffic UT is thus $50/24=2.08\%$ of the network load; for a high-traffic UT, the generated load is $50/8=6.25\%$ of the network load, i.e., three times as much as a low-traffic UT. The load generated by each UT is split equally among the UT's three switched Poisson input traffic sources; each source's λ_H and λ_L values used to obtain the desired network loads, and the corresponding values of λ given by Equation (5) (Section 3.5), are presented in

Table 2 below. Note that these values are per traffic source; since each UT has three sources, the given λ values must be multiplied by three to obtain the mean traffic rate generated by each UT.

Total Load (%)	λ (cells/s)		λ_L (cells/s)		λ_H (cells/s)	
	Low-Traffic UT	High-Traffic UT	Low-Traffic UT	High-Traffic UT	Low-Traffic UT	High-Traffic UT
95	63.73	191.19	21.24	63.73	106.22	318.66
90	60.38	181.13	20.13	60.38	100.63	301.89
85	57.02	171.07	19.01	57.02	95.04	285.12
80	53.67	161.01	17.89	53.67	89.45	268.34
75	50.31	150.94	16.77	50.31	83.86	251.57
70	46.96	140.88	15.65	46.96	78.27	234.80
60	40.25	120.75	13.42	40.25	67.09	201.26
50	33.54	100.63	11.18	33.54	55.90	167.71

Table 2: Traffic source parameters for various network loads

4.1.2.2. *SRD vs. LRD traffic*

As noted in Section 3.3, the LRD and self-similar properties in network traffic have significant implications for queuing performance: LRD input traffic in queuing systems generally results in longer queues (and thus, increased queuing delays and packet losses) than SRD traffic [30]. Recall that in this thesis, SRD and LRD traffic is modeled by MMPP and PMPP sources, respectively; these two source types are differentiated by their state sojourn time distributions – exponential for MMPP, and Pareto for PMPP. Simulations are run with both source types in order to compare the resulting DCA performance. To provide a fair basis for comparison, we use the same mean state sojourn time for both source types, 3.25 seconds as stated in Section 4.1.1.

The PMPP traffic's self-similarity is measured by the Hurst parameter H ; self-similar traffic with $0.5 < H < 1$ is also long-range dependent, with H values closer to 1 resulting in burstier traffic [38]. The Hurst parameter is given by [38]:

$$H = \frac{3-c}{2} \quad 1 < c < 2 \quad (19)$$

In our simulations, H is set to 0.8, hence $c = 1.4$; since the mean state sojourn time $E(x) = 3.25$ seconds as stated above, Equation (3) gives $a = 0.929$.

4.1.2.3. Moving-average sliding window size

Recall from Section 3.4 that L is the size of the moving-average sliding window used by each UT for the estimation of the mean traffic arrival rate in each MAC queue. With a smaller window size, the moving average covers a shorter time span (fewer frames) and thus, follows the instantaneous traffic arrival rate more closely, responding more quickly to traffic transients and to state changes of the switched Poisson input traffic source. This implies high variation in the estimated mean. As L is increased, the estimated mean becomes smoother, i.e., the estimate is of better quality, possibly leading to improved DCA performance; the downside is that the estimate responds more slowly to traffic variations such as the traffic source's state switching point, with the opposite effect of reducing DCA performance. This tradeoff between the estimate's quality and its responsiveness is investigated by running simulations with $L = 1, 5, 10, 19$ and 80 frames (corresponding to $0.0265, 0.1325, 0.265, 0.5035$ and 2.12 seconds, respectively, with the frame time of 26.5 ms given in Section 4.1.1).

4.1.2.4. Request transmission threshold

Recall from Section 3.6.3 that the request transmission threshold Δ_{TH} is used by each UT to determine whether to transmit a capacity request to the scheduler in the current frame. Since Δ_{TH} allows the DCA signaling overhead to be adjusted, it is the parameter used to investigate the performance-signaling tradeoff: simulations are run for $\Delta_{TH} = 0.0, 0.2$ and 0.5 (0, 20 and 50%). $\Delta_{TH} = 0.0$ signifies that the UT sends a capacity request whenever there is any change in the request amount, regardless of the magnitude of this change; $\Delta_{TH} = 0.0$ is thus the reference case used in the evaluation of the DCA performance degradation resulting from decreasing the signaling overhead.

4.2. Results and discussions

4.2.1. Comparison of DCA cases

Plots of the average UT cell queuing delays vs. the normalized network load are used to evaluate and compare the overall behavior, performance and robustness of the proposed DCA cases at various network loads ranging from light to heavy (50 to 95%), and with both input short-range dependent MMPP and long-range dependent PMPP traffic types.

Figure 11 shows the average queuing delays for a typical low-traffic UT using the baseline DCA case C1 (Section 3.6.1) with MMPP and PMPP input traffic. The service differentiation between the EF, AF and DE queues is clearly visible, with the EF queue having the lowest queuing delays and the DE queue, the highest (as expected); this is because the EF queue receives the largest share of the capacity assigned by the scheduler, and the DE queue, the smallest share. The queuing delays increase with the network load, which is typical of most multiple-access schemes because the level of contention between users is higher at higher loads. As the load decreases, the queuing delays for all three queues appear to converge towards a value between 0 and 0.05 seconds; the convergence value is actually half the frame time, $26.5/2=13.25$ ms. This convergence occurs because at low network loads, the UT queues are usually empty, so the capacity request amounts are small and the scheduler thus assigns most of the MF-TDMA frame slots using FCA. Consequently, each UT is assigned slots in every frame, regardless of whether any were requested, so a cell arriving in a UT queue can usually be transmitted immediately (in the current frame). The cell's waiting delay in this case can be considered as a random variable uniformly distributed between 0 and the frame time $T_F = 26.5$ ms, i.e., on average, the cell's waiting delay is equal to half the frame time, 13.25 ms as noted above.

In Figure 11, we observe that the queuing delays with PMPP input traffic are higher than with MMPP traffic. This reinforces the well-known result that LRD input traffic in queuing systems generally results in longer queues (and thus, increased packet queuing delays) than SRD traffic [30].

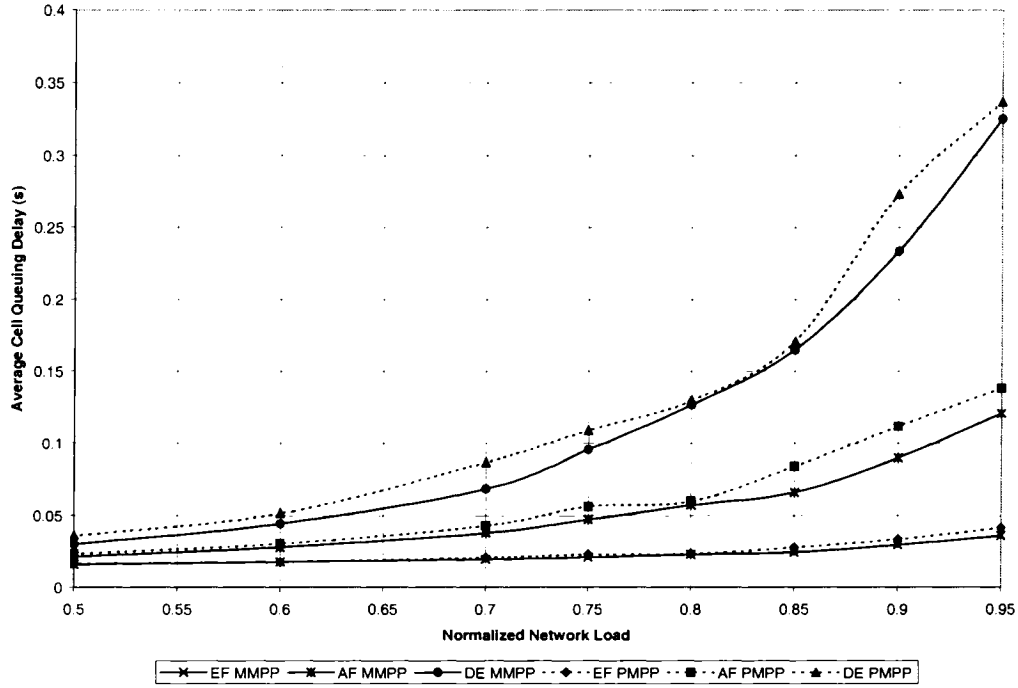


Figure 11: Average cell queuing delay vs. network load for low-traffic UT, case C1, MMPP and PMPP traffic

Next, we consider the queuing delay differences between low-traffic and high-traffic UTs. Figure 12 shows the average cell queuing delays vs. network load for typical low and high-traffic UTs using the baseline DCA case C1. We note that the queuing delays for the low and high-traffic UTs are similar up to a network load of 85%, above which the delays for the high-traffic UTs are higher, especially in the cases of the AF and DE queues. These higher delays are not caused by the DCA scheme, but rather by a design limitation of the UTs: recall from Section 2.1.2 that a UT cannot transmit on multiple carriers simultaneously, in order to reduce the complexity and cost of the transmitter; consequently, each UT's maximum transmission rate is limited to the carrier rate, 512 kb/s as stated in Section 4.1.1. At 95% network load, each traffic source of a high-traffic UT generates traffic with a mean rate $\lambda = 191.19$ cells/s, as shown in Table 2 (Section 4.1.2.1). Since there are three such sources per UT (one for each MAC queue), the mean traffic rate generated by the UT is

$$191.19 \text{ cells/s/traffic source} * 3 \text{ traffic sources} * 53 \text{ bytes/cell} * 8 \text{ bits/byte} = 243 \text{ kb/s}$$

This rate represents 47% of the carrier rate. Conversely, the mean traffic rate generated by a low-traffic UT is 81 kb/s, only 16% of the carrier rate. Hence, the low-traffic UT has more *transmission headroom* available than the high-traffic UT for handling high-intensity traffic bursts (transients), resulting in lower queuing delays. For network loads not exceeding 85%, both UT types have sufficient headroom available so that the carrier rate ceases to be a limiting factor, resulting in comparable queuing delays as shown in Figure 12.

Recall that the present chapter is concerned with evaluating the performance of the proposed DCA scheme. As described above, the carrier capacity limit skews the performance results in a somewhat artificial manner not due to the operation of the scheme itself, but rather to a design limitation of the UT hardware; the capacity limit can thus obscure the DCA performance evaluation and comparison. Consequently, in our subsequent discussions, we only consider the low-traffic UT because it has sufficient transmission headroom available, even at high network loads, so that the carrier capacity is not a performance-limiting factor, allowing the true DCA performance to be more clearly observed.

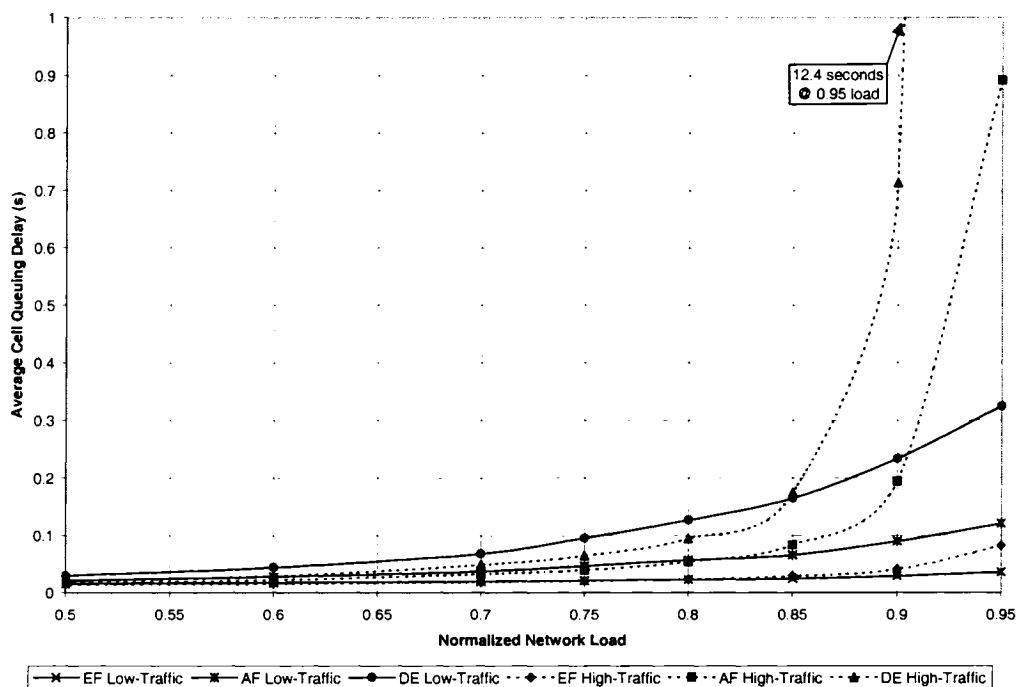


Figure 12: Average cell queuing delay vs. network load for low and high-traffic UTs, case C1, MMPP traffic

Figure 13 compares the average queuing delays for the three proposed DCA cases (C1, C2 and C3). For C2 and C3, we set the moving-average sliding window size $L = 19$ frames, corresponding to 0.50 seconds (see Section 4.1.2.3); C1 does not use a moving average. For C3, we set the request transmission threshold $\Delta_{TH} = 0.0$ (see Section 4.1.2.4), signifying that each UT sends a capacity request whenever there is any change in the request amount, regardless of the magnitude of this change; this is done to ensure a fair comparison with C1 and C2, which do not use a request transmission threshold.

First, we note that the average queuing delays of C1 and C2 are nearly identical. Recall that in C1 (Section 3.6.1), we assumed that the mean traffic rates λ_H and λ_L of each MAC queue's input traffic source are known by the UT, and that the UT can detect the traffic source's state changes, allowing the required capacity C_{ik} to be recalculated whenever a state change occurs, using Equation (14) with the λ value corresponding to the new state (λ_H or λ_L). In C2 (Section 3.6.2), on the other hand, λ in Equation (14) is *estimated* using a moving time-average of cell arrivals. Since the queuing delays of C1 and C2 are nearly identical, we therefore conclude that estimating λ (rather than assuming it known) does not impair the DCA performance; this result is significant because in a practical BSA network, λ *must* be estimated – it is unrealistic to simply assume it known.

Thus far, we have compared the performance of C1 and C2; we now introduce C3 into our discussion. For the EF queue, C3 provides nearly identical queuing delays as the other two DCA cases. For the AF and DE queues, however, the delays provided by C3 are higher than for the other cases as the network load increases. The delay difference between C3 and the other cases is very small for the AF queue, but more pronounced for the DE queue; in fact, the only case in which C3 provides significantly higher queuing delays than C1 and C2 is for the DE queue at network loads exceeding 75% (see Figure 13). By definition, however, the DE queue represents best-effort service without QoS guarantees, so the higher delays in this case are not of major concern, especially since these higher delays only occur at relatively high network loads as noted above. Now, recall from Section 3.6.3 that C3's scheduling algorithm is significantly simpler

than that used for C1 and C2, reducing C3's complexity; given the minor importance of the resulting performance penalty as described above, C3 is thus the best choice in terms of the performance-complexity tradeoff. In addition, C3's request transmission threshold mechanism is used to reduce the required signaling overhead (this reduction is quantified in Section 4.2.2.2); furthermore, C3's capacity requests contain less information than in the other two cases, further reducing the signaling overhead. Consequently, C3 is also the best choice in terms of the performance-signaling tradeoff. Among our three proposed DCA cases, C3 thus provides the best balance between queuing performance, algorithmic complexity and signaling overhead requirements; consequently, we focus our subsequent discussions on C3.

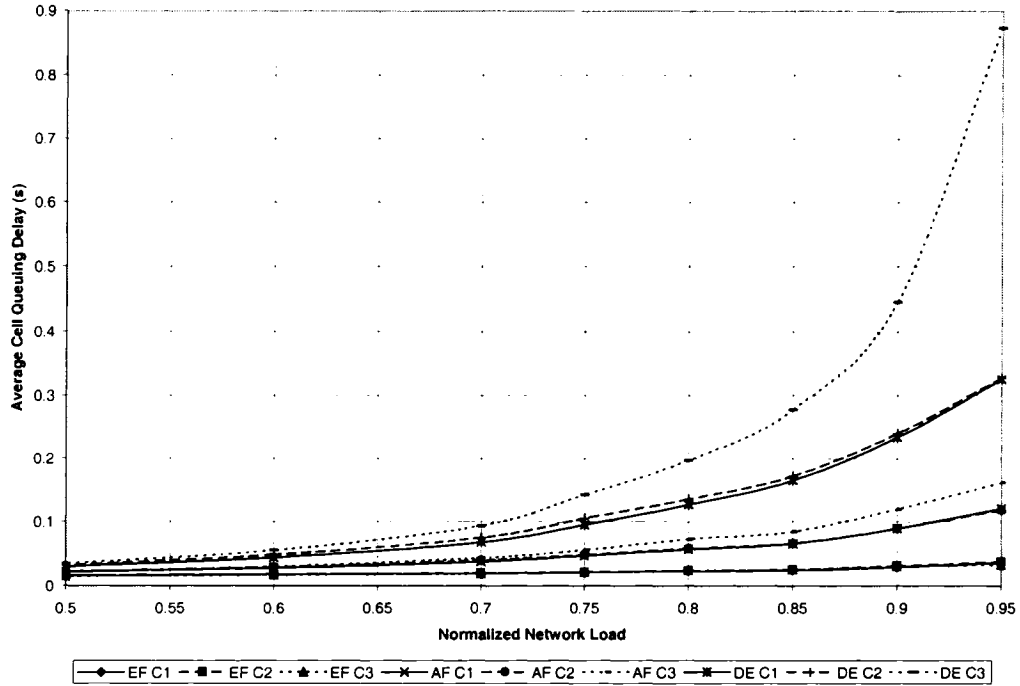


Figure 13: Average cell queuing delay vs. network load for a low-traffic UT, cases C1-C3, MMPP traffic

4.2.2. QoS performance ($\Pr\{Q>n\}$)

As noted in Section 3.1, the queue size survival function $\Pr\{Q>n\}$ is a convenient measure of the QoS experienced by cells in the UT's MAC queues. In this section, plots of the UT queue size survival functions for case C3 are presented.

These plots are used to determine whether the QoS criterion of Equation (1) (Section 3.1) is satisfied for each MAC queue, and to optimize the DCA parameters described in previous sections, i.e., the moving-average sliding window size L and request transmission threshold Δ_{TH} . High network loads ranging between 80 and 95% are used for the plots because, as noted previously, performance differences and trends are most apparent when the network is heavily loaded since the DCA is more heavily stressed.

Figure 14 shows the queue size survival functions at 95% network load with MMPP and PMPP input traffic. The moving-average sliding window size L is set to 19 frames and the request transmission threshold Δ_{TH} is set to 0.0, as in all previous Figures. We note that for all three queues, the PMPP curves decay more slowly than the MMPP curves, indicating higher probabilities of obtaining larger queue sizes with PMPP input traffic; this finding parallels the results obtained in [38], as well as the average queuing delay results obtained in Section 4.2.1 of the present thesis (PMPP traffic was found to result in higher queuing delays). This behavior is due to the long-range dependence and self-similarity of the PMPP traffic.

Next, we note that each survival function curve in Figure 14 appears to be composed of two regions: for small queue sizes (below approximately 10 cells), the slope of the curve is steeper than for larger queue sizes. This dual-slope shape is most apparent for the EF queue, but is also visible for the two other queues. The shape is characteristic of queuing systems with switched (modulated) traffic arrival processes, like those used in this thesis. The first region (with the steeper slope) is called the *cell region*, and the other, the *burst region* [41]. To facilitate our discussion of these regions, recall from Section 3.1 that the queue size survival function can be interpreted as the cell loss probability of a queue with finite length n . For small n (i.e., in the cell region), the loss is due mainly to the cell variability of the Poisson traffic arrival process within each state of the switched traffic source. In the cell region, increasing n allows this variability to be absorbed by the larger queue size, so the cell loss rapidly decreases as indicated by the steep slope of the survival function. As n is further increased, the

burst region is entered and the loss due to cell variability becomes negligible, resulting in a decreased slope of the survival function. In the burst region, the cell loss is caused mainly by the input traffic source's transitions into its "high" state, because the new mean cell arrival rate λ_H momentarily exceeds the queue service rate, leading to a queue build-up. The DCA process causes the service rate to be increased to compensate for the new, higher arrival rate, but this only occurs after the DCA cycle time lag described in Section 2.1.4.

We now use Figure 14 to determine whether the QoS criterion of Equation (1) (Section 3.1) is satisfied for each of the UT's MAC queues, with the n_k queue size values specified in Section 4.1.1 ($k = 1, 2, 3$ denotes the EF, AF and DE queue, respectively): $n_1=75$, $n_2=115$ and $n_3=150$ cells, and the outage probability $P_{OUT}=10^{-3}$. For the EF queue, we note that at the specified queue size $n_1=75$ cells, $\Pr\{Q>n_1\}$ with both MMPP and PMPP traffic is well below the specified outage probability, hence the QoS criterion is satisfied for the EF queue despite the very high network load used in Figure 14 (95%). The QoS criterion is violated for the AF and DE queues, however, i.e., the outage probability requirement is not met at the specified queue sizes $n_2=115$ and $n_3=150$ cells, respectively.

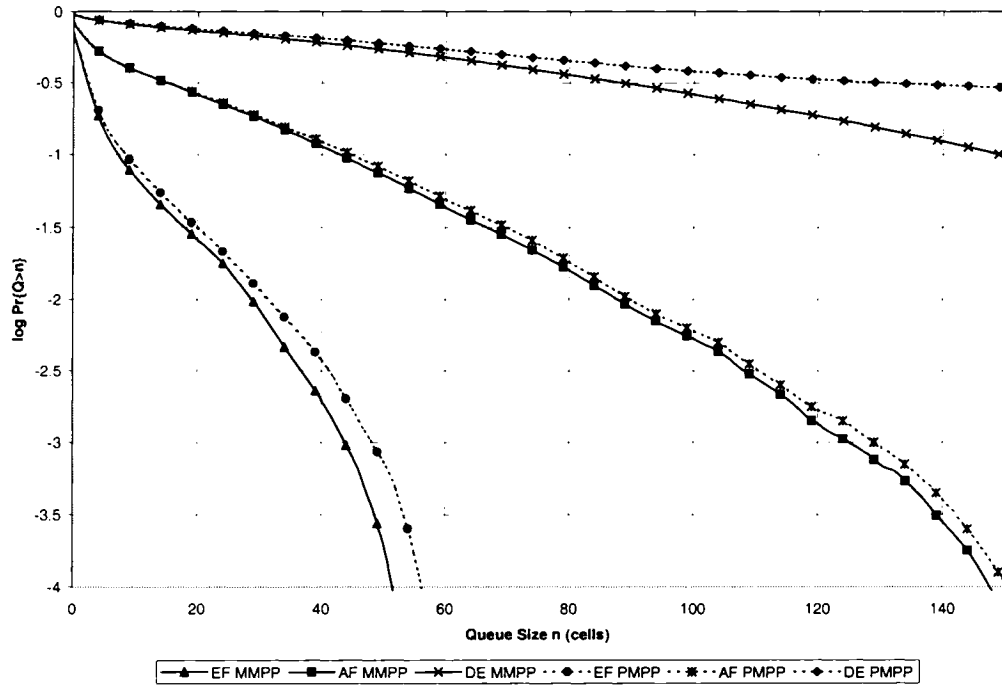


Figure 14: Queue size survival functions, 95% load, case C3, MMPP and PMPP traffic

In order to determine the maximum network load at which the outage probability requirement is met for the AF and DE queues, Figure 15 shows $\Pr\{Q > n_k\}$ (with n_2 and n_3 as above) for these two queues vs. the network load. We note that for the AF queue, the outage requirement is met at loads not exceeding approximately 92% with MMPP traffic and 90% with PMPP traffic; similarly, for the DE queue the outage requirement is met at loads not exceeding approximately 87% with MMPP traffic and 84.5% with PMPP traffic. Thus, with PMPP input traffic, lower network loads are required than with MMPP traffic in order to obtain the same QoS.

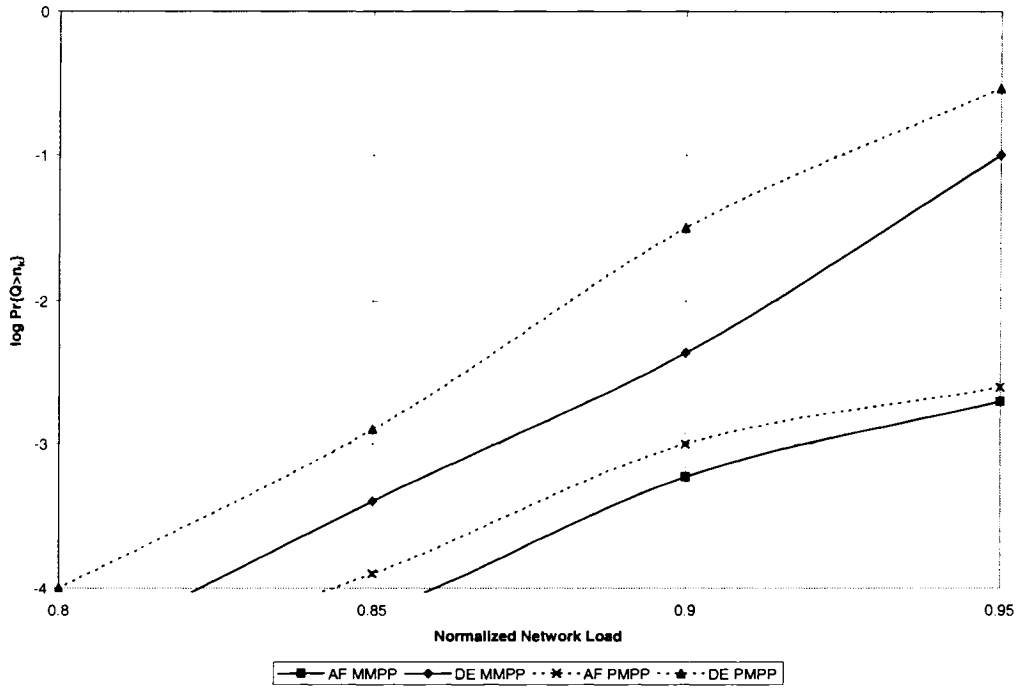


Figure 15: $\Pr\{Q > n_k\}$ ($n_2=115$, $n_3=150$) vs. network load, case C3, MMPP and PMPP traffic

4.2.2.1. Effects of the moving-average sliding window size

We now consider the effects of the moving-average sliding window size L . Recall from Section 3.4 that with a smaller L , the moving average used to estimate the mean traffic arrival rate in each UT MAC queue covers a shorter time span (fewer frames) and thus, follows the instantaneous traffic arrival rate more closely, responding more quickly to traffic transients and to state changes of the

queue's switched Poisson input traffic source. This implies high variation in the estimated mean. As L is increased, the estimated mean becomes smoother, i.e., the estimate is of better quality, possibly leading to improved DCA performance; the downside is that the estimate responds more slowly to traffic variations such as the traffic source's state switching point, resulting in slower updates of the capacity assigned to the UT by the scheduler, with the opposite effect of reducing DCA performance. There is thus a tradeoff between the estimate's quality and its responsiveness to input traffic variations.

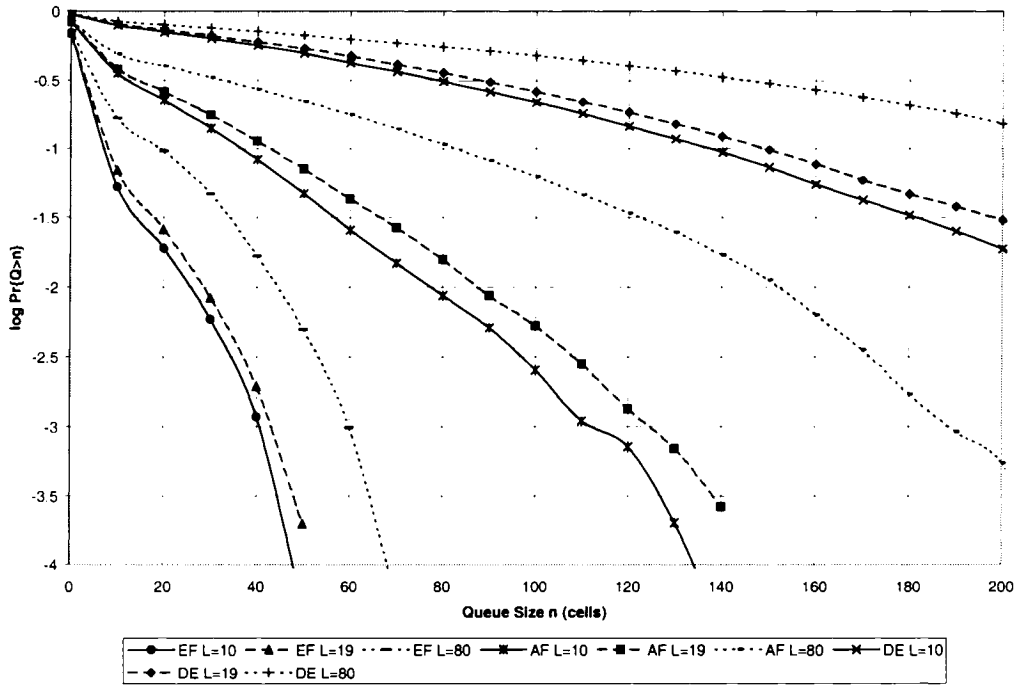


Figure 16: Queue size survival functions, 95% load, case C3, MMPP traffic, $L=10,19,80$ frames

Figure 16 and Figure 17 show the queue size survival functions for $L = 10, 19, 80$ and 1, 5, 10 frames, respectively. The effects of changing L are clearly observable: for all three queues, lower L values result in survival functions that decay more quickly, implying smaller queue sizes and thus, better performance. In terms of the estimate quality-responsiveness tradeoff described above, we therefore conclude that the estimate's improved quality at higher L values is outweighed by its reduced responsiveness to input traffic variations. Hence,

lower values of L are preferable, subject to diminishing returns: decreasing L from 19 to 10 frames results in improved performance for all three queues, but a further decrease from 10 to 5 frames only results in further slight improvement for the AF and DE queues. The final decrease from 5 frames to 1 results in slight improvement for the DE queue, but not the EF and AF queues; in fact, we note that for these two queues, better performance is actually obtained with $L = 5$ than with $L = 1$ over certain queue size ranges (though the differences are very small). We therefore conclude that the good window size is in the range of 5 to 10 frames.

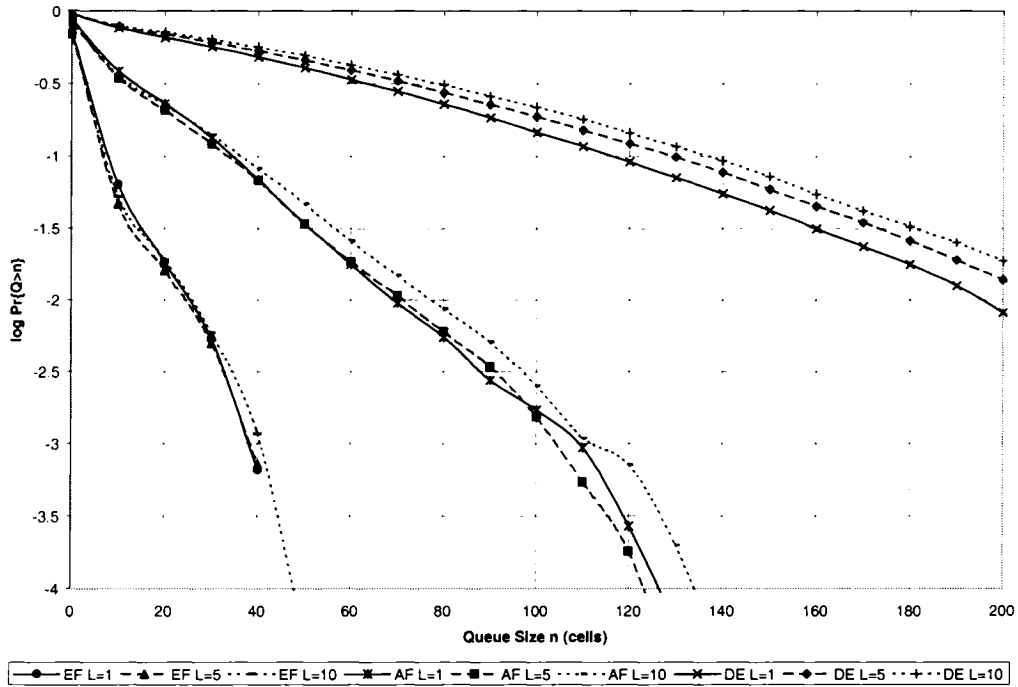


Figure 17: Queue size survival functions, 95% load, case C3, MMPP traffic, $L=1,5,10$ frames

4.2.2.2. Effects of the request transmission threshold

Recall from Section 3.6.3 that the UT's request transmission threshold mechanism is used to determine whether a capacity request should be transmitted, based on the relative change in the request amount since the previous request. The request transmission threshold Δ_{TH} is the minimum relative change required to transmit a request. As Δ_{TH} is increased, the amount of signaling used for DCA decreases because fewer capacity requests are transmitted per unit time. The tradeoff is that

the scheduler must rely on older request information for the capacity assignment process; depending on the Δ_{TH} setting, this older information may no longer reflect the current input traffic conditions at the UT, resulting in sub-optimal capacity assignments, and thus, decreased queuing performance. In other words, the DCA becomes less responsive to input traffic variations as Δ_{TH} is increased. A tradeoff (compromise) is thus required between DCA performance and signaling overhead. Δ_{TH} is the parameter used to investigate this tradeoff.

Figure 18 shows the queue size survival functions for $\Delta_{TH} = 0.0, 0.2$ and 0.5 . We note that generally, lower values of Δ_{TH} result in survival functions that decay more quickly, implying smaller queue sizes and thus, better performance. We emphasize, however, that the observed differences are small. Increasing Δ_{TH} from 0.0 to 0.2 results in no performance penalty for the EF queue, and only minimal penalties for the AF and DE queues; a further increase to 0.5 results in a minimal performance penalty for the EF queue as the queue size n increases, and slightly more substantial penalties for the AF and DE queues. Overall, the EF queue is thus almost unaffected by Δ_{TH} changes.

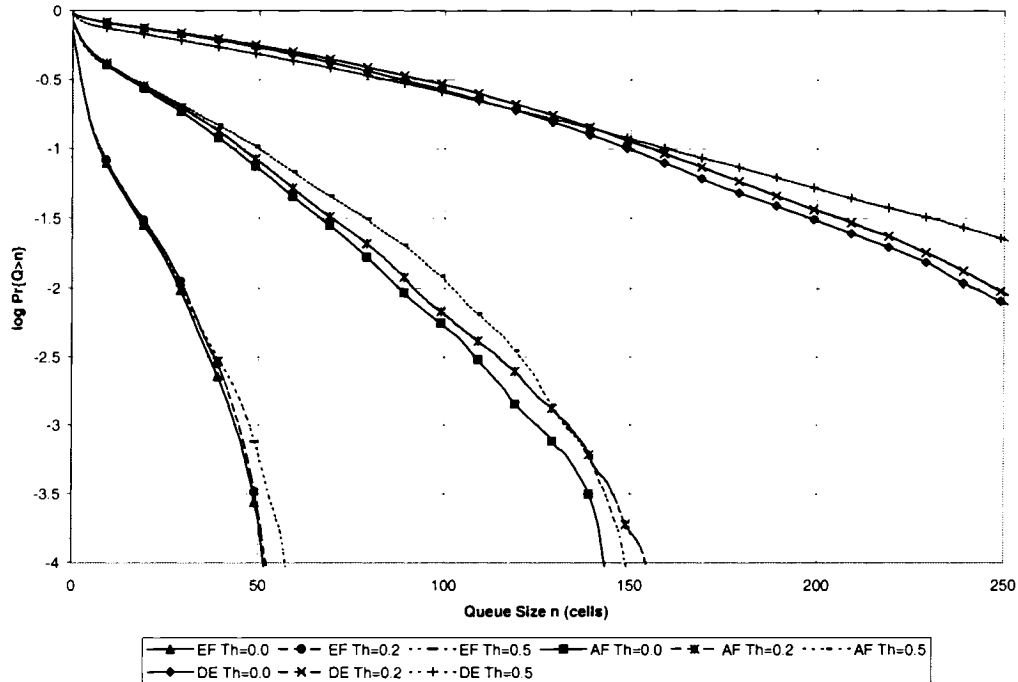


Figure 18: Queue size survival functions, 95% load, case C3, MMPP traffic, $\Delta_{TH}=0.0,0.2,0.5$

Now, let us consider the signaling overhead with the above Δ_{TH} values. The signaling overhead is measured/quantified by the proportion of frames in which capacity requests are transmitted for each queue (service class): a lower proportion signifies less signaling; a proportion of 1.0 indicates that a request is transmitted in every frame. Table 3 shows these proportions for the above Δ_{TH} values. First, we note that for each Δ_{TH} value, the request proportions for the three queues are nearly identical, which is understandable since the input traffic sources used for the queues are homogeneous (i.e., all have the same parameters), so the resulting traffic in each queue has similar characteristics. Next, we note that even with $\Delta_{TH}=0.0$, the request proportion is less than 1.0, signifying that requests are not transmitted in every frame, because the request amount is sometimes identical from frame to frame (depending on the traffic arrival pattern). Since the request proportions for the three queues are nearly identical as noted above, we use the EF queue for illustrative purposes. We note that the request proportion decreases from 0.755 to 0.102, representing an 86.5% reduction in the amount of signaling, by increasing Δ_{TH} from 0.0 to 0.2; by contrast, a signaling reduction of 96.4% is achieved by increasing Δ_{TH} from 0.0 to 0.5, i.e., only an additional 10% reduction is achieved by the increase to 0.5.

Δ_{TH}	Requests per frame		
	EF Queue	AF Queue	DE Queue
0.0	0.755	0.755	0.754
0.2	0.102	0.102	0.106
0.5	0.027	0.028	0.027

Table 3: Capacity requests per frame with multiple Δ_{TH} values, MMPP traffic

In summary, increasing Δ_{TH} from 0.0 to 0.2 results in no queuing performance penalty for the EF queue and only minimal penalties for the AF and DE queues, while reducing the signaling overhead by 86.5%; slightly more substantial performance penalties result from further increasing Δ_{TH} to 0.5, while only reducing the signaling by an additional 10%. We therefore conclude that $\Delta_{TH}=0.2$ is a good choice in view of the performance-signaling tradeoff.

4.2.3. Network overload

Recall from Section 3.5 that a network overload occurs when the aggregate traffic arrival rate from all the UTs in an uplink beam momentarily exceeds the beam's total transmission capacity (although the aggregate *mean* arrival rate does not), because too many switched Poisson input traffic sources are simultaneously in their “high” states. During network overloads, the traffic arrival rates to the UT queues thus exceed the queue service rates because there is insufficient transmission capacity available in the affected uplink beam; the queues therefore become unstable (build-up), resulting in increased queue sizes and delays. Consequently, the overloads' occurrence frequency (probability) should be low, and their duration should be short, in order to minimize the performance impact.

Table 4 shows the network overload probability and mean network overload time (duration) at 95% load, obtained both analytically and by simulation. The overload probability derivation presented in Section 3.5.1 is independent of the state sojourn time distribution of the switched Poisson input traffic sources, and is thus valid for both MMPP and PMPP traffic. We note that the simulation overload probability values are in close agreement with the analytical value. In addition, the simulation values for MMPP and PMPP traffic are also in close agreement with each other, which is unsurprising since, as noted above, the overload probability should not depend on the state sojourn time distribution.

	Analytical	Simulation	
		MMPP	PMPP
Overload Probability	0.238	0.249	0.243
Mean Overload Time	0.25435 sec	0.2880 sec	0.3031 sec

Table 4: Simulation and analytical network overload probabilities and mean network overload times at 95% load

The mean overload time derivation presented in Section 3.5.2 uses Markov analysis, and is thus only valid for MMPP input traffic, with its exponentially distributed state sojourn times. From Table 4, we note that the analytical value is in close agreement with the simulation value for MMPP traffic. In addition, the simulation values for MMPP and PMPP traffic are also close, despite the different state sojourn time distributions of these traffic types. This is because the *mean*

state sojourn times are the same for both traffic types (as noted in Section 4.1.2.2), resulting in the same mean time elapsed before traffic sources in the “high” state switch to the “low” state, thus ending the overload.

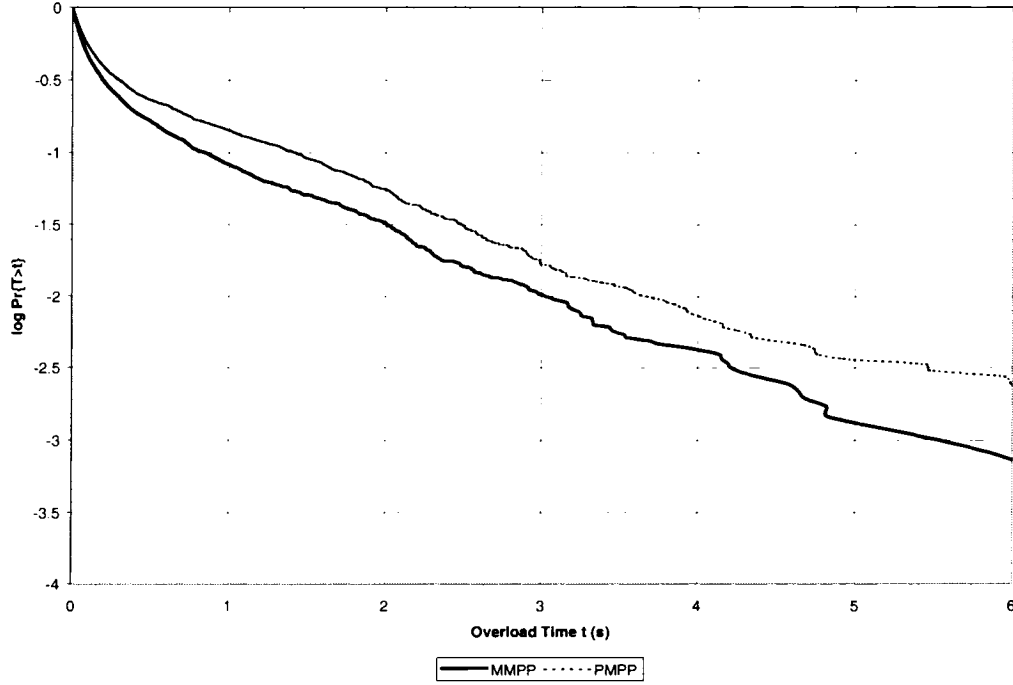


Figure 19: Network overload time survival functions, 95% load, MMPP and PMPP traffic

We now consider the survival function of the overload time. Figure 19 shows the survival functions with MMPP and PMPP input traffic. We note that the MMPP curve decays more quickly than the PMPP curve, indicating higher probabilities of obtaining longer overload times with PMPP traffic (although the *mean* overload time remains the same, as noted above). This behavior is due to the heavy-tailed Pareto state sojourn time distribution used for PMPP traffic: longer sojourn times are obtained with a higher probability than in the case of distributions that are not heavy-tailed (e.g., the exponential distribution used for MMPP traffic); consequently, longer network overload times are also obtained with a higher probability because more time elapses before traffic sources in the “high” state switch to the “low” state, thus ending the overload. The effects of these longer overload times on the UT’s queuing performance were observed in the simulation results of Sections 4.2.1 and 4.2.2: increased queuing delays and

queue sizes with PMPP traffic, as compared to MMPP traffic. In particular, it was found in Section 4.2.2 that with PMPP traffic, the network load must be reduced in order to maintain the same QoS performance as with MMPP traffic.

4.2.4. Effects of traffic state switching rate on DCA performance

So far, it was assumed that the mean state sojourn times of the switched Poisson input traffic sources are greater than the DCA cycle time; this is referred to as *slow-switching* (SS) traffic. This assumption allowed us to treat the traffic within a DCA cycle as simple (non-switched) Poisson traffic, since each traffic source was assumed to remain in the same state for the entire duration of a DCA cycle, as described in Section 3.3. This simplification guided the derivation in Section 3.6.1 of the service rate (capacity) required for each of the UT's MAC queues in order to satisfy the QoS criterion of Equation (1) (Section 3.1); the results of this derivation played a key role in the design and operation of our proposed DCA scheme. We now consider *fast-switching* (FS) input traffic; in this case, the mean state sojourn times of the traffic sources are *smaller* than the DCA cycle time, so it is no longer possible to make the simplifying assumption that each traffic source remains in the same state for the entire duration of a DCA cycle. Simulation is used to determine how FS traffic affects the DCA performance, and thus, to ascertain whether the DCA scheme must be adjusted for operation with FS traffic. Recall from Section 4.1.1 that the DCA cycle time is 0.54 seconds, and that for SS traffic, the mean state sojourn times of the traffic sources in the “high” and “low” states, $1/r_H$ and $1/r_L$, respectively, are both set to 3.25 seconds. Now, for FS traffic, we set the mean sojourn times to 0.5, 0.05 or 0.005 seconds (for comparison). Figure 20 and Figure 21 show the queue size survival functions at 95% network load with SS and FS (with mean sojourn time of 0.5 seconds) MMPP and PMPP input traffic, respectively. We note that for all three queues, the FS curves decay more quickly than the SS curves, indicating smaller queue sizes (better performance) with FS traffic. To further illustrate this trend, Figure 22 shows the survival functions with SS and FS (mean sojourn times = 0.5, 0.05 and 0.005 seconds) MMPP traffic; for comparison, simple (non-switched) Poisson

traffic is also included. To avoid cluttering the figure, only the curves for the EF queue are shown, but the performance trend is the same for the other two queues.

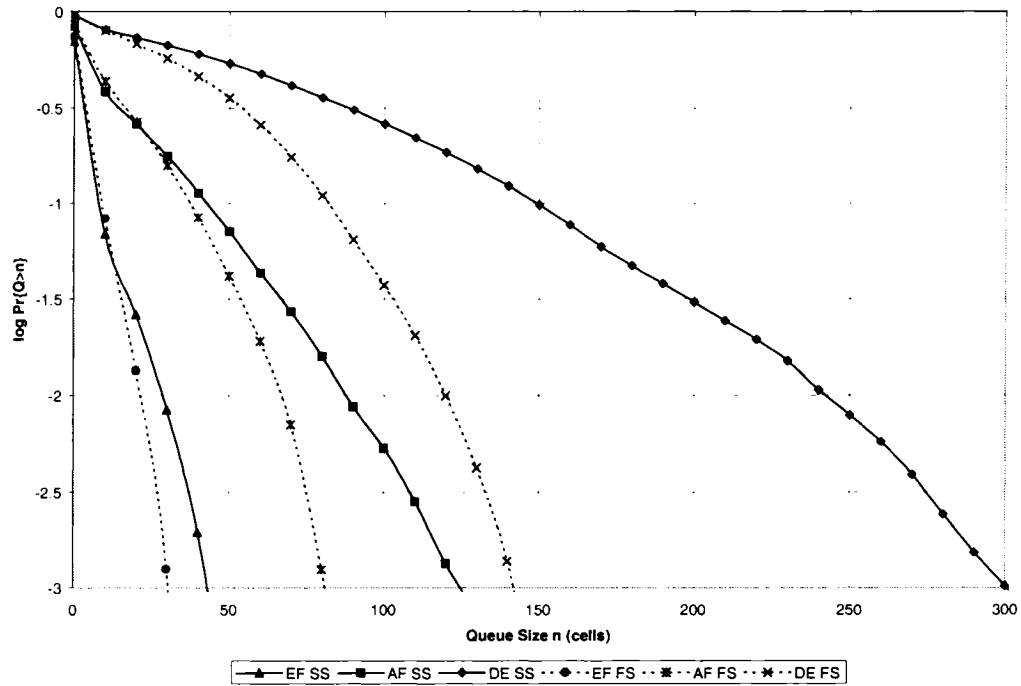


Figure 20: Queue size survival functions, 95% load, case C3, SS and FS (mean sojourn time = 0.5 sec) MMPP traffic

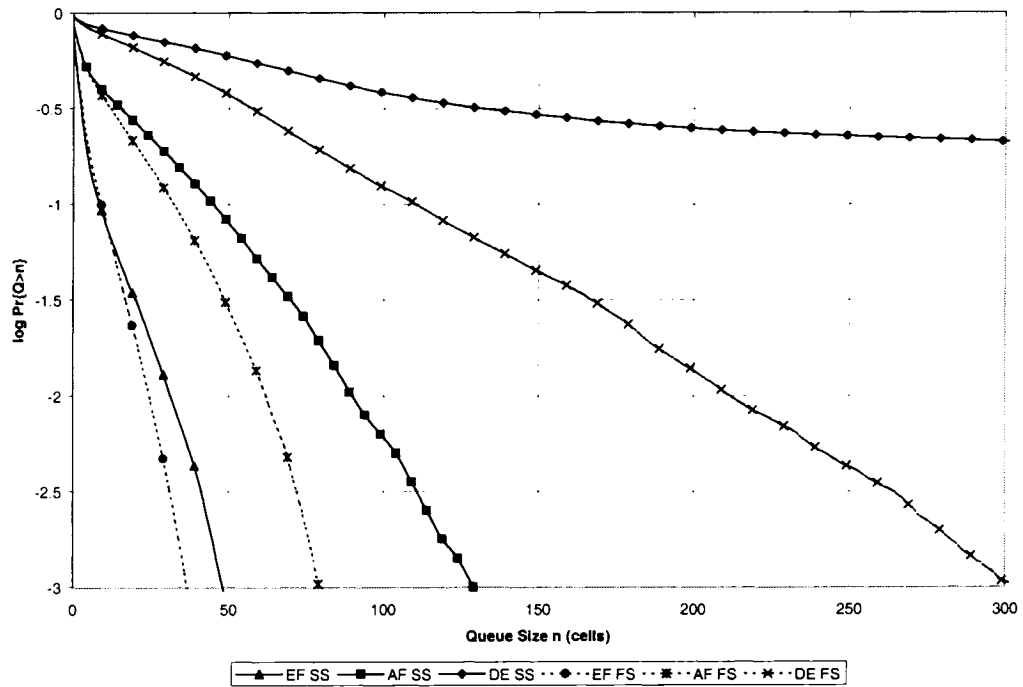


Figure 21: Queue size survival functions, 95% load, case C3, SS and FS (mean sojourn time = 0.5 sec) PMPP traffic

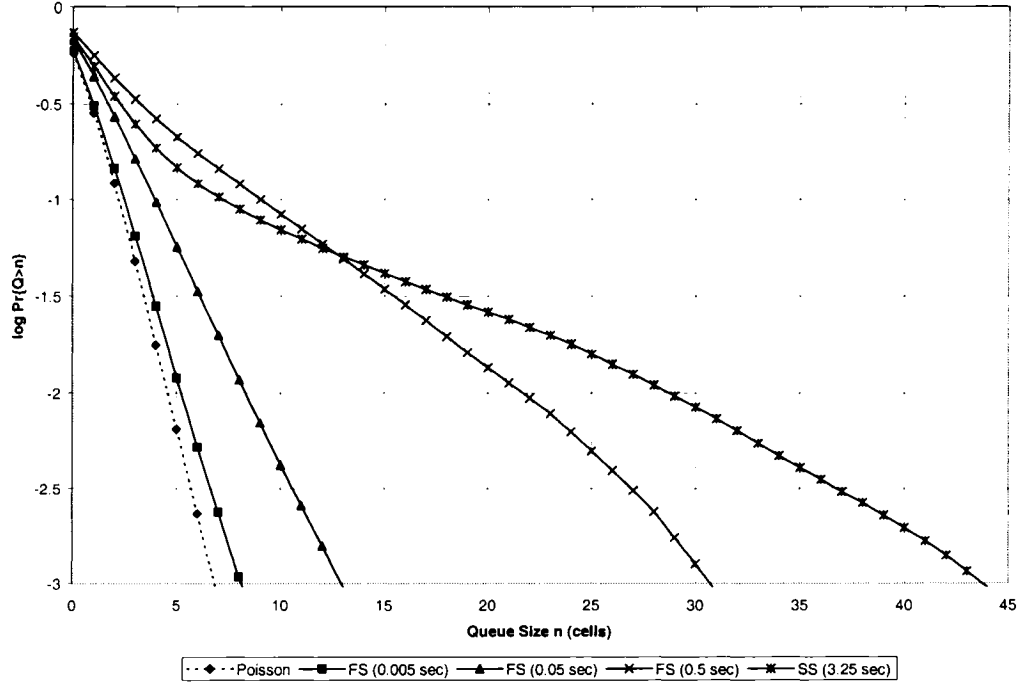


Figure 22: EF queue size survival functions, 95% load, case C3, SS and FS MMPP and Poisson traffic

In Figure 22, the best performance (fastest decay of the survival function) is obtained with the Poisson traffic input. As the mean state sojourn time of the MMPP traffic is decreased, the overload time and hence the queuing delay due to the burst region (see Section 4.2.2) are also reduced, and the MMPP survival function curve approaches the Poisson one. For the mean sojourn time of 0.005 seconds, the MMPP and Poisson survival function curves are almost identical. It is also noticed that the survival function curve for slow-switching MMPP (with the mean sojourn time of 3.25 seconds) clearly exhibits two slopes corresponding to the cell and burst regions (as discussed in Section 4.2.2), while the curves for fast-switching MMPP (especially with the mean sojourn time of 0.005 seconds) seem to have only one slope corresponding to the cell region, as in the case of Poisson traffic.

Figure 23 shows the probability density function (PDF) curves for Poisson and MMPP traffic models. The PDF of slow-switching MMPP (with the mean sojourn time of 3.25 seconds) exhibits two peaks corresponding to the mean cell rates of the “low” and “high” states, $\lambda_L=21.24$ and $\lambda_H=106.22$ cells/s (as shown in

Table 2, Section 4.1.2.1). As the mean state sojourn time is decreased, we note that the shape of the PDF gradually changes to that of the Poisson PDF with mean rate of 63.73 cells/s (as given by Equation (5), Section 3.5).

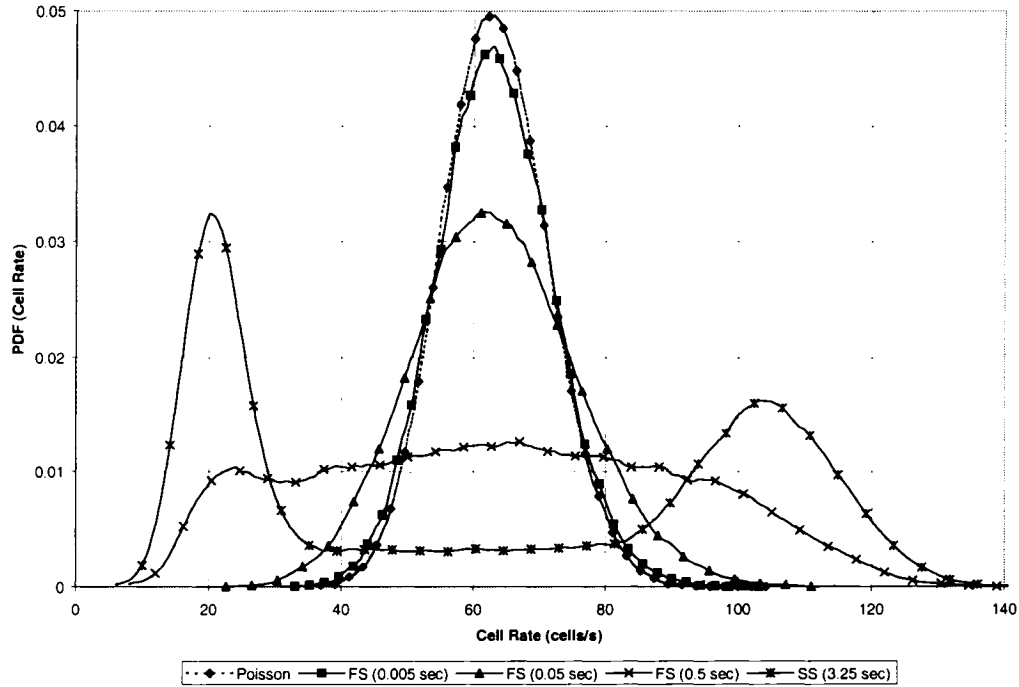


Figure 23: Cell arrival rate PDF, 95% load, SS and FS MMPP and Poisson traffic

Recall from Section 3.6 that the design of our DCA scheme was influenced by the assumption that with SS input traffic, the traffic within a DCA cycle time can be treated as simple (non-switched) Poisson traffic. As shown by our above simulation results, this assumption still holds with FS traffic, because the PDF of the switched Poisson traffic arrival process approaches the Poisson PDF as the mean state sojourn time decreases, implying that the switched traffic becomes equivalent to Poisson traffic. We therefore conclude that it is not necessary to adjust/modify our DCA scheme for operation with FS traffic.

Chapter 5 : SUMMARY AND CONCLUSIONS

The main theme/objective of this thesis has been to propose a novel dynamic capacity allocation (DCA) scheme for IP-based broadband satellite access (BSA) networks. The proposed scheme should be efficient with regard to satellite bandwidth utility and satisfy user QoS requirements, while maintaining reduced signaling overhead and computational complexity. The work and main conclusions of the thesis are summarized below.

In Chapter 2, we provided the background/context and reference BSA system architecture necessary for presenting our DCA-related research in subsequent thesis chapters. We began with a survey/summary of previous work in BSA network architectures and current architectural trends, including the DCA concept. Next, we described the current state of the two IP QoS architectures in terrestrial networks, IntServ and DiffServ, noting that DiffServ is currently the preferred architecture due to complexity and scalability concerns regarding IntServ. Finally, we described our proposed architectures for a DiffServ-capable scheduler and user terminal (UT).

In Chapter 3, we presented our proposed DCA scheme, which aims to provide high satellite bandwidth utilization and satisfy user QoS requirements, while maintaining reduced signaling overhead and algorithmic complexity. We began by defining an appropriate QoS performance measure, i.e., one that can be used to derive the basic QoS parameters of packet delay and loss. We then provided a survey/summary of previous work on DCA schemes for satellite networks. Next, we presented the input traffic model used in the BSA network, justifying its validity using results from the literature and assumptions regarding the typical operational environment of the UTs. We continued by describing the moving-average estimator of the mean traffic arrival rate, a key DCA input. Next, we described the network overload situation and its expected negative performance impact, setting the stage for our subsequent derivation of the network overload probability and mean overload time. Finally, we presented our DCA scheme. A key component of the scheme is an analytical expression for the capacity required

by each UT to achieve a target QoS for the user traffic; this expression was derived using an analytical queuing model based on the previously selected input traffic model. Our DCA scheme was presented in the form of three *study cases*, each in itself a self-contained, operational DCA scheme. These three cases represent a logical evolution of the capacity request and assignment algorithms, allowing us to evaluate and compare multiple DCA design options that were in the end consolidated under the third (final) case. Thus, the first two cases were intended only as intermediate steps towards the final case. Our subsequent simulation results in Chapter 4 showed that this final case represents the best compromise (tradeoff) between QoS performance, signaling overhead and algorithmic complexity.

In Chapter 4, we used mainly simulation (but also some analytical) results to evaluate the performance of the proposed DCA scheme. Simulation was used for the optimization of various DCA elements/parameters; this optimization included the selection of the capacity request and assignment algorithms (in view of the performance-complexity tradeoff), the traffic averaging window size (in view of the tradeoff between the estimate's quality and its responsiveness to input traffic variations), and the request transmission threshold (in view of the performance-signaling tradeoff). We began by describing the simulation model's configuration and selecting appropriate simulation scenarios to evaluate the desired aspects of the DCA performance and facilitate the aforementioned optimization. Next, we presented and analyzed the simulation results (comparing them to analytical results where appropriate), drawing conclusions regarding the DCA optimization. Finally, we considered the case of fast-switching input traffic, as opposed to the slow-switching traffic considered hitherto: simulation was used to determine how fast-switching traffic affects the DCA performance, and thus, whether the proposed DCA scheme must be adjusted for operation with fast-switching traffic. The conclusions established in Chapter 4 are summarized below:

- DCA algorithm selection: Among the three proposed DCA cases, C3 provides the best balance in terms of the tradeoff between queuing performance and algorithmic complexity: incurring only a minimal performance penalty

relative to the other two DCA cases, C3 employs a significantly simplified scheduling algorithm. In addition, a request transmission threshold mechanism allows C3's signaling overhead to be reduced; C3's capacity request messages contain less information than those in the other two DCA cases, further reducing the signaling overhead. For all these reasons, C3 is the preferred DCA case.

- Choice of moving-average sliding window size L : The good range for the moving-average sliding window size is 5-10 frames. Values above this range lead to decreased queuing performance because the estimated mean traffic arrival rate is less responsive to input traffic variations (in particular, the traffic source's state switching point) for higher values of L . On the other hand, values below this range cause no further improvement to the queuing performance, and can actually impair it due to excessive variation in the estimated mean, thus reducing the estimate's quality.
- Choice of request transmission threshold Δ_{TH} : In view of the tradeoff between the DCA performance and the signaling overhead required by the DCA scheme, the good value for the request transmission threshold Δ_{TH} is 0.2. This setting causes a very small queuing performance degradation (as compared to the case of $\Delta_{TH}=0.0$), while reducing the signaling overhead by 86.5%. Increasing Δ_{TH} beyond 0.2 results in decreased queuing performance with no further significant reduction of the signaling overhead.
- Effects of correlated traffic: LRD input traffic in the BSA network results in longer network overload times than with SRD traffic. The effect of these longer overload times is reduced queuing performance (increased delays and queue sizes), due to inadequate queue service rates during the overload periods. To maintain the same queuing performance with LRD traffic as with SRD traffic, it is therefore necessary to reduce the network traffic load.
- Effects of traffic state switching rate: Fast-switching input traffic results in better queuing performance than slow-switching traffic, because it becomes equivalent to simple (non-switched) Poisson traffic. The design of our DCA scheme was influenced by the assumption that in the slow-switching case, the

traffic within a DCA cycle can be treated as Poisson. Since this assumption also holds in the fast-switching case, we conclude that it is not necessary to make any adjustments/modifications to our proposed DCA scheme for operation with fast-switching traffic.

In this thesis, the request transmission threshold Δ_{TH} was selected by statically setting it to various values and using simulation to compare the resulting queuing performance and signaling overhead. In a practical BSA network, UT queuing performance *feedback* could be used to dynamically adjust Δ_{TH} in response to changing queuing conditions. As the number of packets in each queue increases (i.e., as the queue builds up), Δ_{TH} could be decreased in order to increase the request frequency, and thus, the DCA scheme's responsiveness to input traffic variations, leading to improved queuing performance (decreased queue length) because the assigned capacity would more closely follow the input traffic. For instance, the queue could conceptually be divided into “green”, “yellow” and “red” regions of operation corresponding to increasing queue lengths; the Δ_{TH} setting would then depend on the queue's current region, with “red” corresponding to the lowest setting (highest request frequency) and “green”, the highest setting (lowest request frequency). Alternatively, the queue's $\Pr\{Q > n\}$ could be continually monitored, and if the outage probability threshold P_{OUT} is crossed (violated) too often, Δ_{TH} could be decreased.

The UT queuing performance may be further improved by implementing a distributed congestion control scheme with the participation of both the scheduler and the UTs, using feedback from the scheduler regarding the global network congestion level: in any given MF-TDMA frame, if the sum of all requested capacity in the network (from all the UTs) exceeds the frame's capacity (i.e., if there is no free capacity available), the scheduler assumes that the network is congested and signals this to the UTs by setting a flag in the broadcasted slot assignment map. Upon receiving this congestion indicator, each UT then decreases its throughput by performing more aggressive connection admission control (e.g., refusing all new connections) and in extreme cases, possibly even

dropping existing ones. While this congestion control policy would decrease the traffic volume carried by the UT, it may allow the QoS performance of existing connections to be maintained at acceptable levels even during periods of network congestion.

The above issues, involving the use of queue length feedback to dynamically adjust the request frequency, and of global congestion level feedback for congestion control, could potentially lead to improved DCA performance. These issues are left for future researchers to explore.

REFERENCES

- [1] Postel J. RFC 791: Internet Protocol, 1981.
- [2] Braden R, Clark D, Shenker S. RFC 1633: Integrated Services in the Internet Architecture: an Overview, 1994.
- [3] Blake S *et. al.* RFC 2475: An Architecture for Differentiated Service, 1998.
- [4] Grossman D. RFC 3260: New Terminology and Clarifications for Diffserv, 2002.
- [5] Le-Ngoc T, Elshabrawy T. Broadband satellite access for interactive multimedia services. *Space Communications* 2001; **17**:35-48.
- [6] ETSI EN 301 790: Digital Video Broadcasting (DVB), Interaction channel for satellite distribution systems.
- [7] Le-Ngoc T. Dynamic resource allocation schemes for multimedia satellite communications. *The 4th International Symposium on Personal, Indoor and Mobile Radio Communications (RIMRC)* 1993; 552-556.
- [8] Evans B. *Satellite Communication Systems*; Institution of Electrical Engineers (IEE), 1999.
- [9] Verma D. *Supporting Service Level Agreements on IP Networks*; Que, 1999.
- [10] Wang Z. *Internet QoS: Architectures and Mechanisms for Quality of Service*; Morgan Kaufmann Publishers: San Francisco, 2001.
- [11] Kota S, Marchese M. Quality of service for satellite IP networks: a survey. *International Journal of Satellite Communications and Networking* 2003; **21**(4-5):303-349.
- [12] Braden R *et. al.* RFC 2205: Resource Reservation Protocol (RSVP) – Version 1 Functional Specification, 1997.
- [13] Wroclawski J. RFC 2210: The Use of RSVP with IETF Integrated Services, 1997.
- [14] Herzog S. RFC 2750: RSVP Extensions for Policy Control, 2000.
- [15] Shenker S, Partridge C, Guerin R. RFC 2212: Specification of Guaranteed Quality of Service, 1997.

- [16] Wroclawski J. RFC 2211: Specification of the Controlled-Load Network Element Service, 1997.
- [17] Handley M *et. al.* RFC 2543: SIP: Session Initiation Protocol, 1999.
- [18] Davie B *et. al.* RFC 3246: An Expedited Forwarding PHB (Per-Hop Behavior), 2002.
- [19] Heinanen J *et. al.* RFC 2597: Assured Forwarding PHB Group, 1999.
- [20] Le-Ngoc T, Mohammed J.I. Combined free/demand assignment multiple access (CF-DAMA) protocols for packet satellite communications. *IEEE International Conference on Universal Personal Communications (ICUPC)* 1993; **2**:824-828.
- [21] Ha T.T. *Digital Satellite Communications*; McGraw-Hill Publishing Company, 1990.
- [22] Mohammed J.I, Le-Ngoc T. Performance analysis of combined free/demand assignment multiple access (CFDAMA) protocol for packet satellite communications. *IEEE International Conference on Communications (ICC)* 1994; **2**:869-873.
- [23] Le-Ngoc T, Krishnamurthy S.V. Performance of combined free/demand assignment multi-access (CFDAMA) protocol with pre-assigned request slots in integrated voice/data satellite communications. *IEEE International Conference on Communications (ICC)* 1995; **3**:1572-1576.
- [24] Le-Ngoc T, Krishnamurthy S.V. Performance of combined free/demand assignment multiple-access schemes in satellite communications. *International Journal of Satellite Communications* 1996; **14**(1):11-21.
- [25] Le-Ngoc T, Jahangir I.M. Performance analysis of CFDAMA-PB protocol for packet satellite communications. *IEEE Transactions on Communications* 1998; **46**(9):1206-1214.
- [26] Mitchell P.D, Grace D, Tozer T.C. Performance of the combined free/demand assignment multiple access protocol with combined request strategies via satellite. *IEEE Personal, Indoor and Mobile Radio Communications Conference (PIMRC)* 2001.
- [27] Mitchell P.D, Grace D, Tozer T.C. Comparative performance of the

- CFDAMA protocol via satellite with various terminal request strategies. *IEEE Global Telecommunications Conference (GLOBECOM)* 2001; 4:2720-2724.
- [28] Jiang Z, Leung V.C.M. A Predictive Demand Assignment Multiple Access Protocol for Internet Access over Broadband Satellite Networks. *International Journal of Satellite Communications and Networking* 2003; 21(4-5):451-467.
 - [29] Farmer J.J, Sidrowich J.J. Predicting chaotic time series. *Physical Review Letters* 1987; 59:845-848.
 - [30] Erramilli A, Narayan O, Willinger W. Experimental Queuing Analysis with Long-Range Dependent Packet Traffic. *IEEE/ACM Transactions on Networking* 1996; 4(2):209-223.
 - [31] Leland W.E., Wilson D.V. High time-resolution measurement and analysis of LAN traffic: implications for LAN interconnection. *IEEE Infocom* 1991; 1360-1366.
 - [32] Fowler H.J., Leland W.E. Local area network traffic characteristics, with implications for broadband network congestion management. *IEEE Journal on Selected Areas in Communications* 1991; 9(7):1139-1149.
 - [33] Leland W.E., Taqqu S.M., Willinger W, Wilson D.V. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking* 1994; 2(1):1-15.
 - [34] Heffes H., Lucantoni D.M. A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. *IEEE Journal on Selected Areas in Communications* 1986; 4(6):856-868.
 - [35] Shah-Heydari S., Le-Ngoc T. MMPP modeling of aggregated ATM traffic. *Proc. Canadian Conference on Electrical and Computer Engineering (CCECE)* 1998; 129-132.
 - [36] Subramanian S.N., Le-Ngoc T. Traffic modeling in a multi-media environment. *Proc. Canadian Conference on Electrical and Computer Engineering (CCECE)* 1995; 838-841.

- [37] Fischer W., Meier-Hellstern K. The Markov-modulated Poisson process (MMPP) cookbook. *Performance Evaluation* 1992; **18**:149-171.
 - [38] Le-Ngoc T, Subramanian S.N. A pareto-modulated Poisson process (PMPP) model for long-range dependent traffic. *Computer Communications* 2000; **23**(2):123-132.
 - [39] Huang A, McDonald D. Connection admission control for constant bit rate traffic at a multi-buffer multiplexer using the oldest-cell-first discipline. *Queuing Systems* 1998; **29**(1):1-16.
 - [40] Leon-Garcia A, Widjaja I. *Communication Networks: Fundamental Concepts and Key Architectures*; McGraw-Hill, 2000.
 - [41] Shroff N.B., Schwartz M. Improved Loss Calculations at an ATM Multiplexer. *IEEE/ACM Transactions on Networking* 1998; **6**(4):411-421.
-