Combined Free/Demand Assignment Multiple Access Protocols for Packet Satellite Communications

Jahangir I Mohammed

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Applied Science at
Concordia University
Montreal, Quebec, Canada

June 1993

ISBN 0-315-87323-X

ABSTRACT

Combined Free/Demand Assignment Multiple Access Protocols for Packet Satellite Communications

Jahangir I Mohammed

The number of information processing terminals are increasing at a rapid pace. A need to exists to organize these geographically dispersed terminals into a network in order exchange information and share processing capabilities. A geosynchronous satellite provides a benign broadcast/multi-access medium to achieve this networking. A new class of protocols suitable for such a satellite network is proposed. Two versions of the proposed class of protocols are simulated, and analyzed within the queuing theoretic framework. The performance of proposed protocols are evaluated in terms of three performance measures: average packet delay, variance of packet delay and cumulative probability distribution of packet delay. Cumulative probability distribution is a new measure introduced here that gives information on the reliability in meeting a certain delay constraint. Comparison with other pertinent existing schemes shows the proposed schemes to be eminent in terms of all the above performance measures for a wide range of user population size.

DEDICATION

To my parents Mrs. Fathima Ibrahim and Mr. Mohammed Ibrahim ,

with love and reverence

ACKNOWLEDGEMENTS

It has been a pleasure for me working under the abled supervision of Dr. Tho LeNgoc, whose guidance and encouragement have been invaluable throughout the course of this research. I gratefully acknowledge his support.

I thank all my professors; but for them, it would not have been possible for me to carry through this research. I thank Sukima and my friends for giving a warm and friendly atmosphere that was conducive to fruitful research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The number of information processing terminals are increasing at a rapid pace. There exists a need to organize these terminals into a network in order to exchange information and share processing and storage capabilities. A geosynchronous satellite provides a benign broadcast/multi-access medium to interconnect these geographically distributed information processing terminals. The attractiveness of satellite medium for this purpose is largely due to the recent advancements that the satellite technology has witnessed [17] [9]: Today's satellites are more sophisticated in design, larger in size and power. The earth stations, in turn, reflect the reverse trend. They are smaller, less complex and are much cheaper. Now receive only micro-terminals are available small enough to be placed on the top of a desk. This drastic decrease in the size and cost of earth stations have made possible for the earth-stations to be placed right at the users premises making direct networking of information processing terminals plausible. Besides the cost effectiveness, the satellite medium also offers a number of other advantages [19]:

1. Elimination of complex topological design
2. wide geographical coverage
3. Accommodation of mobile users, and

4. flexibility

As the number of terminals continue to proliferate, each contributing a small fraction of the total traffic, the method of organizing the channel access so as to achieve an efficient channel utilization becomes a complex problem. This issue is further complicated by the fact that the data traffic generated by the terminals are often bursty i.e., short spurts of data interspersed with relatively long random idle periods come with a *small delay constraint*. Packet switching has proven to be the most efficient way of sharing channel capacity among a number of bursty data sources. When terminals are dispersed over wide geographical area and connected through multi-access/broadcast channel, an efficient multiple access technique is essential for the co-ordination of often conflicting channel demands in packet switched networks.

This thesis inquires methods of achieving efficient sharing of the satellite resources among a large number of geographically dispersed information terminals.

1.1 Communication Satellites

The basic characteristics and the constraint associated with the use of satellite to provide digital communication networks are considered in this section. This provides a framework and background for the rest of the work in this thesis.

1.1.1 The Medium

Figure 1.1 is a block diagram of a satellite communication system [16]. The satellite communication link has several important characteristics from user point of view: data rate, error rate, and propagation delay. Link data rates could range from a few hundred bits per second to hundreds of megabits per second depending on the application and choice of the system parameters. For fixed power and bandwidth parameters on the RF link through the satellite, a trade off is possible between the

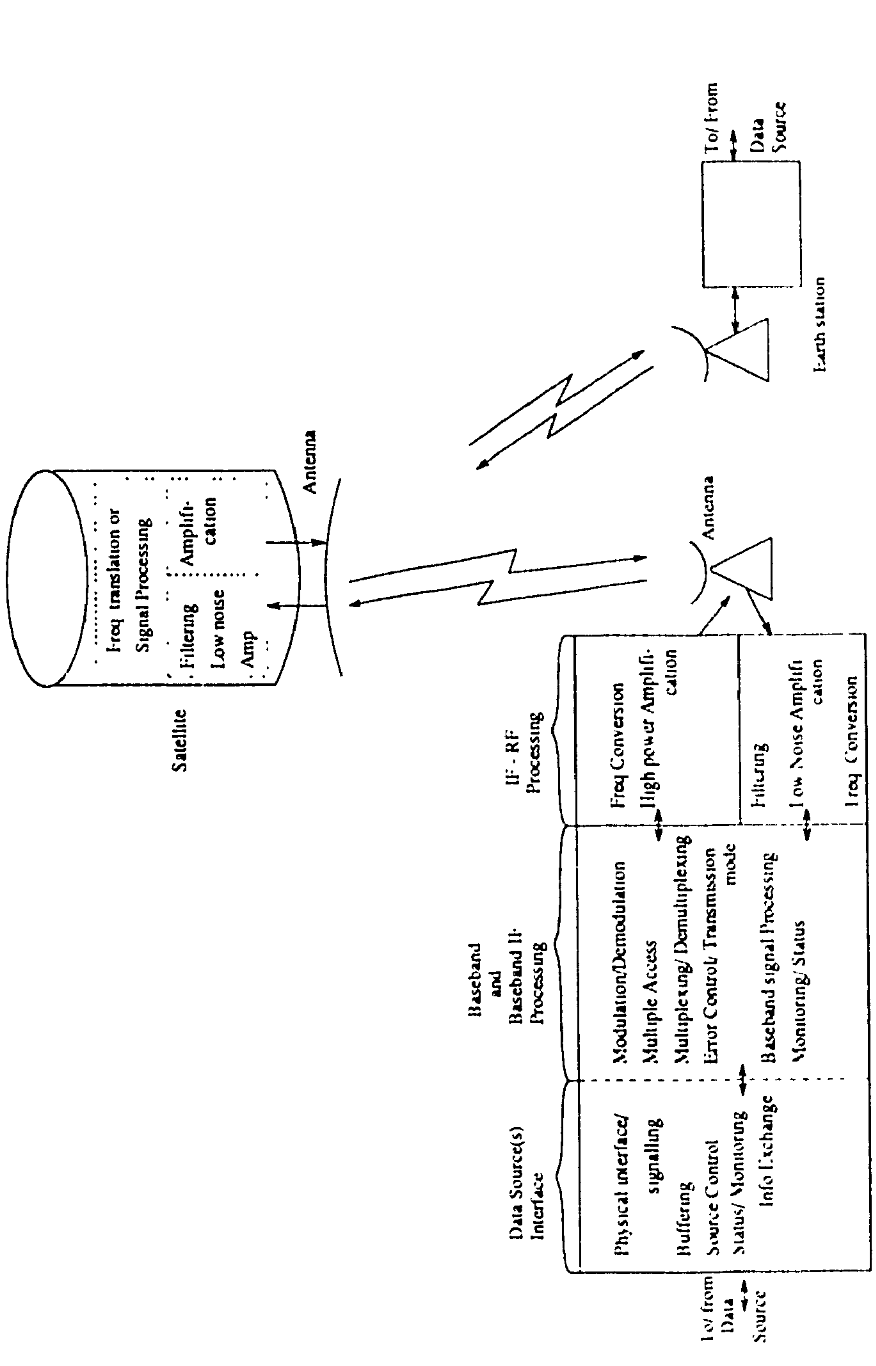


Figure 1.1: Block diagram model of satellite communication system

link data rate and its error performance; decreases in the data rate typically permit better error performance. Forward error correction techniques provide an efficient way to improve error performance with fixed power by reducing the data rate or increasing the bandwidth. The round trip propagation delay i.e., the propagation time from one earth station to another via the satellite, is approximately 270 millisecond.

The satellite can be viewed as a centralized power and bandwidth resource which must be shared among a number of earth stations (user terminals). An important characteristic of a satellite communication system is its coverage. This refers to the portion of the earth's surface from which an earth station can access the satellite and from which earth stations can receive the satellite transmissions. A satellite can provide broadcast capability at any given time to all earth stations within its transmission coverage area. The combination of the multiple access and broadcast capabilities has an important networking implication: it is possible to form the earth stations into a fully connected one-hop network.

1.1.2 Channelization

Division of the overall satellite resources (power and bandwidth) among the various earth station accesses (earth station uplink transmissions) occurs at three levels:

Transponder Channelization

Most present day satellites contain a set of independent transponders each operating in different frequency bands. Each transponder contains an input filter to limit reception to a selected region of the uplink frequency band followed by a frequency translator, power amplifier, and output filter to limit transmission to a specified region of the downlink frequency band. Thus the overall satellite power and bandwidth is divided up on a frequency basis.

Data rate reduction

A typical transponder supports data rates of hundreds of megabits per second. The basic channelization (multiple-access) techniques are applied to derive a number of reduced data rate channels. This data rate reduction is required to reduce the cost of the receiver. The multiple access techniques such as FDMA and TDMA are used for this purpose. For FDMA, channelization is achieved by deriving the total transponder bandwidth into a number of frequency-subbands and allocating one subband to each access subgroup. In the case of TDMA, channelization is achieved by temporal sharing of the entire bandwidth and power among various accesses. This temporal sharing involves the use of a frame structure and implies a requirement for maintaining global timing among the earth stations and use of burst, as contrasted to continuous communication techniques.

Multiple access in reduced data rate channels

The derived low data rate channel should be efficiently shared among a number of geographically dispersed user terminals. This is the major subject of this thesis, discussed extensively in later chapters.

1.1.3 Networking

The multiple access and broadcast capabilities of a satellite can be used to form a network from a set of cooperative earth stations in a number of ways. Some realizations involve fixed capacity allocations on links connecting the earth stations, other realizations involve dynamic sharing to provide the capacity and connectivity required by fluctuating traffic demands. In systems which utilize dynamically changing connectivity or capacity, the response time required to change the connectivity or capacity allocations strongly influences the properties of network realization.

Three basic types of satellite network realizations illustrate some of the options and tradeoffs involved. These realizations are: (i) dedicated channel, (ii) dedicated uplink/broadcast-downlink, and (iii) shared-uplink/shared-downlink. In dedicated channel realization, fixed data rate point to point links connect each pair of stations in the network (assuming fully connected network). For N-node network, this realization requires $N(N-1)$ full duplex links, with each link sized to handle the average point-to-point traffic between the two connected stations while satisfying a suitable utilization or traffic delay constraint.

In a dedicated-uplink/broadcast-downlink realization, each station in the network has a dedicated fixed-data-rate uplink which can be received by all the other stations in the network. This realization requires N-broadcast simplex links for an N-station network, with each link sized to handle the average traffic originating from that station. Each station multiplexes its own originating traffic on its simplex link to transmit this traffic to the other stations in the network.

In the shared-uplink/shared-downlink realization, a single broadcast communication channel is dynamically shared among all the stations using demand assignment techniques to provide the required connectivity and link capacity.

The choice between the use of TDMA and FDMA depends strongly on the network traffic matrix and the method of realization. A shared-uplink/shared-downlink for example, is highly constrained by the use of FDMA. As the network size increases, TDMA becomes more cost effective than FDMA in providing one-hop connectivity because of the less equipment required at the ground stations, although each piece of equipment could be more expensive. TDMA allows more flexibility in establishing and adjusting individual link data rates to the average traffic demands.

This work concentrates on the techniques to realize fully connected networks via shared-uplink/shared-downlink TDMA techniques.

1.1.4 Earth stations

Figure 1.1 shows the basic subsystems of a satellite earth station, namely, antenna, IF-RF processing, baseband and baseband-IF processing and interface. The subsystem requirements, which depend on both the application and the satellite characteristics, play a major role in determining earth stations costs and, hence, the economic feasibility of system concepts which involve large number of earth stations.

The receiving capability of an earth station is typically expressed in terms of the parameter G/T, the gain of the receiving antenna divided by the receiving system noise temperature. Larger values of G/T require larger antennas and/or higher quality low-noise amplifiers with subsequent increase in cost.

Figure 1.1 lists major functions performed in the base-band processing and data source interface subsystems. In a particular system (application) one or more of these functions might be absent or present in a simplified form.

1.2 Motivation behind the Research

A need exists to organize the increasing number of information processing terminals into a network, to exchange information, and to share processing and storage capabilities. The recent advancements in satellite communications have given a benign *high bandwidth* multi-access/broadcast medium to achieve this networking.

The majority of multiple access protocols proposed and analyzed in the literature are designed for a *narrow bandwidth* satellite channel (100 kb/s); their performance suffers when a *wide bandwidth* (2 Mb/s) channel is employed. The present research is concerned with the proposal of a new class of multiple access protocols that is particularly efficient for networking bursty data sources using a high bandwidth (wideband) satellite channel.

1.3 Scope of the Thesis

The contributions of this research can be summarized as follows:

- A new class of Multiple Access Protocols suitable for networking a large number of geographically dispersed information processing terminals using a *Wide-*

band Packet Satellite is proposed.

- Two versions of the proposed class of protocols are computer simulated and analyzed within a queuing theoretic framework.
- The performance of the proposed schemes and a few other existing ones are evaluated, via computer simulation using three performance measures: Average Delay Vs Channel Utilization, Variance of packet delay, and Cumulative probability function of the packet delay. The performances are comprehensively compared.

The rest of the thesis is organized as follows:

Chapter 2 presents an extensive survey of existing multiple access schemes especially applicable to satellite networks. The existing schemes are classified into five categories and traffic environment under which each category performs with superiority is identified. The proposed schemes fall into sixth category. System complexity and implementation issues are discussed in some details.

Chapter 3 describes the proposed class of protocols; its applicability to bent-pipe as well as OBP satellites is discussed. Three versions of the proposed class of schemes are presented, and the pros and cons of each version are discussed.

Chapter 4 analyses two versions of the proposed class of protocols by modeling them as a queuing system. Markov chain analysis and renewal theory are extensively employed. Due to the complexity involved in the analyses, a number of simplifying approximations were made. Then simulations were performed to support the validity of the various approximations. The performances evaluated via analyses are also consolidated by comparing it with simulation results.

Chapter 5 evaluates the performance of the proposed schemes and a few other existing ones in terms of average packet delay, variance of packet delay and cumulative probability distribution of packet delay. The performance is compared in terms of all the above three performance measures for various system parameter values. An intuitive reasoning, for every protocol to exhibit its characters in a certain manner, is also provided.

Chapter 6 gives a summary of the research and suggestions for further work.

Chapter 2

Existing Multiple Access Schemes

The concept of multiple access broadcast communications originates from THE ALOHA SYSTEM of University of Hawaii. Abramson [1] devised a scheme that enabled many computer terminals scattered all over the Hawaiian islands to communicate among each other using a common ground radio channel. He named this scheme, which allowed a *multiple* number of computer terminals to *access* a common communication channel, and still achieve reliable data transfer as ALOHA *multiple access protocol*. *Multiple Access* refers to the capability to share the communication capacity among a population of geographically dispersed users.

In a network of this type, more than one user may attempt to access the common communication channel simultaneously. Simultaneous transmission of more than one message results in message collision, garbling all the messages. What we face is a conflict among users to make exclusive use of the channel for oneself. Apparently, what we need is a procedure to resolve the conflict; in a broad sense this resolution procedure that endows orderly use of channel resource is called the multiple access protocol. The resolution procedure can be *hard*, where the conflict is resolved decisively in such a way that only one user holds the right to access the channel at a time; or, it can be *soft*, where the users abide by a certain set of rules, so as to potentially reduce the conflict (but may not totally eliminate it). In soft procedures, the transmitting users should reliably be able to detect the occasional packet collisions, and retransmit the collided packets. The multiple access protocol can be identified to be located beneath the Data Link Layer of the OSI (Open System Interconnection) reference model [11].

Since the Abramson's ALOHA, a wealth of multiple access protocols of varying degree of 'softness' have been proposed and analyzed in the literature [33]. A primary reason for such abundance is that *different system traffic environment require a different multiple access protocol to achieve efficient sharing of channel resources.*

The most important factor characterizing the multiple access broadcast communications in a distributed environment is the channel propagation delay between users [21]. The role of multiple access protocol is to coordinate the demand of channel access from many geographically distributed users who share a common channel: With multiple access protocol, we organize the separated users into a cooperative queuing structure. The coordination must be carried out through the channel itself. Therefore, the channel propagation delay is directly related to the necessary time it takes for a coordination signal sent by a user to reach all the other users. This is approximately equal to an exorbitant 270 milli-sec for satellite channel; which turns out to be the prime complicating factor in devising an efficient multi access algorithm.

The other important factors that should be taken into consideration in designing a multiple access protocols are user population size of the network and the traffic characteristics (such as length of message, delay tolerance of message, message arrival pattern etc.) of the users.

The prime object in devising a multiple access protocol is - to make most *efficient* use of the channel resource. In literature, the term *efficiency* is almost used synonymous to 'achieving the lowest delay at any given channel utilization[1]'. Justifiably, almost always the Average Message Delay Vs Channel Utilization is used as a

[1]Adhering to convention, throughout this thesis the term efficiency is used in the same meaning

sole criterion to quantify the performance of a multiple access protocol. Practically speaking, though Delay Vs Utilization is the most important performance measure, it alone does not substantially characterize a protocol. The system designer would not be inattentive to a few more characters of the protocol such as (i) variance of delay, (ii) reliability with which a certain delay constraint could be met at a given channel loading (iii) ease of implementation (iv) robustness of the scheme, etc.

In this chapter we present an extensive survey of existing multiple access protocols. Later in chapter 5, we discuss the applicability of these protocols to solve the problem at hand: viz., sharing the satellite channel resource among a large number of geographically distributed bursty data sources. A few prominent protocols including the proposed ones are simulated and comprehensively evaluated in terms of all the above five performance measures. The performance of all the schemes are compared and the eminence of the proposed scheme is established. Guidelines to choose the most suitable protocol for a given system parameters are also presented.

The multiple access protocols proposed so far can be classified into five categories, while the schemes proposed in this work falls into sixth category. Protocols under a particular category use the same idea to achieve their prime object, the 'efficiency'; different protocols within the same category are derived variants of the same idea, to inherit some specific virtue. Each of the categories performs eminently for different traffic environments. In the following, we present for all six categories, the main idea with virtue of which efficiency is achieved, the suitable traffic environment and, prominent example protocols.

The six categories are,

1. fixed Assignment Schemes
2. Pure Random Access Schemes
3. Pure Demand Assignment (Reservation Access) Schemes
4. Combined Random/Reservation Access Schemes

5. Combined Fixed/Demand Assignment Schemes

:

6. Combined Free/Demand Assignment Schemes

The last three categories are dervied by blending the first three categories in a certain way. This is illustrated in Figure 2.1

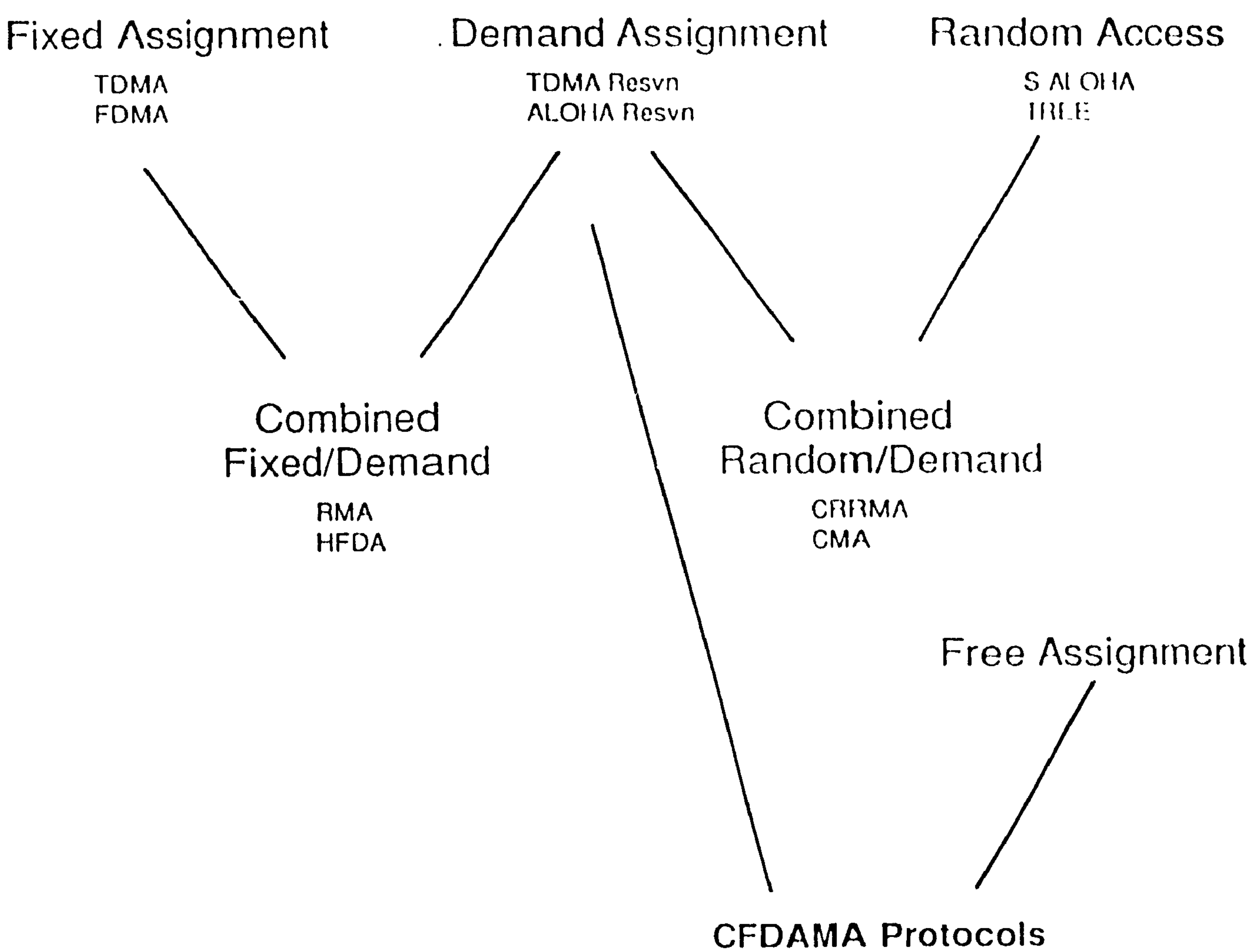


Figure 2.1 The six categories of Multiple Access Protocol

2.1 Fixed Assignment Schemes

The total channel capacity is apportioned in to a number of sub-channels and, one subchannel is permanently allocated to each of the users. In these schemes the channel allocation is fixed and is independent of dynamic activity of user stations. Such fixed allocation is efficient when the number of users is small or when the traffic flow from each user is somewhat streamlined: for systems where the number of users is large, each having randomly occurring infrequent short activity periods with small delay constraints[2], these schemes turns out to be highly inefficient; this idea is further elaborated below for each individual protocol under this category. Based on the way the subchannels are derived, we have four types of fixed assignment schemes: TDMA(Time Division Multiple Access), FDMA(Frequency Division Multiple Access), CDMA(Code Division Multiple Access) and SDMA(Space Division Multiple Access).

In TDMA, the channel time is divided into slots and slots are organized into frames. Slots are permanently assigned to each user in every recurring frame, in proportion to their average traffic rate. A difficulty in TDMA systems is that each station should be synchronized to a global time reference. The global time reference is established either explicitly by a reference station, or implicitly. Since the resources are geographically dispersed, a guard space is required between successive slots in a frame. The accuracy achieved for the timing will determine the required guard space. Accurate synchronization techniques for satellite systems are available to accomplish the global timing with a minimal loss of channel capacity [29]. Further, a certain portion of the frame must also be allocated for frame synchronization. A preamble of about 100 bits is also required with every burst transmission in order for the TDMA modems to acquire frequency, phase, bit timing and framing synchronization. When the number of users is large, a station with a generated message may have to wait for a long time, until it gets its dedicated slot; while the

[2]such users are called *bursty*

unoccupied slots of the other users could still pass by this results in inefficient use of channel when users generate bursty traffic.

In FDMA, the available channel is split into non overlapping frequency bands. One frequency band is permanently allocated to each user, the width of the frequency band being proportional to their average traffic flow. Unlike TDMA no real time co-ordination (synchronization) is required among the accesses (hence can be used to transmit both digital as well as analog signals). Clearly this feature makes FDMA to be the *easiest* to implement. But FDMA suffers from a number of other disadvantages: The transponder must be operated in a near linear mode so that power obtained by each frequency band is in proportion to its up-link power. This requires the satellite power amplifier to operate in the linear mode, which results in some loss of efficiency. Also, the number of users is *limited* by the number of frequency bands into which the total transponder channel capacity can be subdivided which is in-turn *limited* by the intermodulation noise that results from nonlinearities in power amplifiers. The guard space required between adjacent frequency bands is also somewhat more than that required between adjacent slots of TDMA. Finally, TDMA always performs slightly better than FDMA ; I Rubin has shown that the random variable representing the packet delay is always larger in FDMA than in TDMA [31] for comparable systems. Also as explained in chapter 1, FDMA affords greater flexibility in performing changes in the allocation of the bandwidth and network configurations. As in TDMA, for a similar reason, FDMA also turns out to be very inefficient for large bursty user population: many of the frequency bands remain idle while a particular user's transmission takes place at a very low rate compared to the overall bandwidth.

In CDMA scheme, transmissions are allowed to overlap both in frequency as well as in time. This is possible because of the approximate orthogonal property inherent in the pseudo-random codes that are used to modulate the information bits. Multiple orthogonal codes are obtained at the expense of increased bandwidth requirements (in order to spread the waveforms); this also results in lack of flexibil ity in interconnecting all users (unless, of-course, matched filters corresponding to all codes are provided at all receivers).In addition to the orthogonality among the channels, the use of pseudo-random codes may provide message security as well as immunity against jamming. However, the efficiency in power bandwidth utilization in CDMA systems is far lower than those of FDMA or TDMA systems. For this reason, CDMA is not much used in commercial satellite services although widely used in military networks.

In SDMA system, physically separated links (channels) are established. A multi-spot beam satellite system is an example of an SDMA scheme where each spot-beam constitutes a link and each of the geographically non-overlapping spot-beam areas (spot beam zones) is given the entire channel capacity derived from a spot-beam. A problem of capacity sharing among a number of earth stations within a spot-beam zone still exists.

The fixed assignment schemes exhibits the following characteristics, in general:

- The access procedures are conflict free and hence *hard*.
- They are simple, robust and easy to implement.
- They turn out to be highly inefficient when users are bursty and large in number.
- Because of the resource dedication to individual sources, performance is more a function of the number of sources rather than the total traffic generated by the sources.

2.2 Random Access Schemes

As discussed above, fixed assignment schemes are very inefficient for bursty users. This is largely because, the fixed assignment schemes are not adaptive to the large peak to average data transmission ratios, a requisite of bursty users. An approach to achieve large peak to average transmission rate ratio is to provide a single sharable high-capacity channel to all users [35]. The strong law of large numbers then guarantees that with very high probability the demand at any instant will be approximately equal to the sum of the average demands of that population. This idea is the cardinal motivating factor that led to the development of random access schemes. These schemes are suitable when the user population is very large and bursty. It is also easily implemented but the maximum channel utilization is very much limited.

In random access schemes, the entire bandwidth is provided to the users as a single channel to be accessed at will. When more than one user access the channel simultaneously, messages interfere with one another and message (packet) collision occurs. Due to the broadcast property of the channel, the transmitting user can reliably detect the collision after an elapse of maximum station to station propagation. The collided packet can be retransmitted.

A notorious problem in random access schemes is system instability - the feedback loop formed by successive conflicts-retransmissions can cause a runaway condition in the absence of a control technique. Secondly, when station traffic is sufficient to form a queue of new arrivals, transmissions from this queue must also be regulated so as to not increase the probability of a conflict with other stations. For optimum performance, these constraints need to be varied dynamically according to the state of the channel and station queue length [22].

The collision resolution algorithms that achieve system stability has been a subject of considerable study. The simplest of random access schemes, also the first of its kind - pure ALOHA, allows the user to transmit as and when he generates a message with no regard for other users in the system. If the message collides, the station retransmits the message after a *random* delay, in order to avoid repeated collisions. For ALOHA with infinite user population, the maximum channel utilization has been shown to be upper-bounded by 18%. By introducing synchronization in packet transmission, Robert showed an immediate doubling of channel capacity to 36%. Channel time is divided into contiguous slots each long enough to hold a packet and, the beginning of the packet transmission is synchronized to the beginning of a slot. This slotted scheme of Robert has come to be known as S-ALOHA. These bounds on maximum channel utilization can be exceeded if stations are non-homogeneous; i.e., if some stations have significantly higher traffic rate than the others - this effect known as 'excess capacity' can result in efficiency approaching one for certain limiting cases at the expense of many retransmissions per packet for stations with small rates.

Both ALOHA and S-ALOHA are inherently unstable. A number of algorithms has been proposed to make them stable. The frequency of collisions of message is directly proportional to number of stations that are ready to transmit. The idea therefore arose of estimating the number of backlogged stations (say, k) and adapting the retransmission interval to this estimate. This approach cured the problem of instability, while retaining the same throughput. Tsybakov [36] has shown that, if the backlogged stations retransmit after an exponentially distributed random delay, with average delay equivalent to $\frac{1}{(1-e^{-1/k})}$, then the instability disappears. The obvious difficulty with this Tsybakov's *back-off algorithm* is estimating k. Estimates of the number of backlogged stations must be a function of the only information universally available throughout the system: the record of collisions, successful transmissions and idle slots in the succession of transmission slots.

An approach which has provided modest increases in capacity beyond the S-ALOHA's 36% at the expense of some coordination is the tree-based Collision Resolution Algorithms (CRA) [6],[12]. The main idea of these schemes is to waive all the new packets from transmitting until all the collided packets are cleared. The best of these protocols provide capacities in the range of 0.4 to 0.5 and are unconditionally stable [10]. However these algorithms require the knowledge of the outcome of a time slot before the beginning of the next slot. But for satellite channel afflicted with the long round trip propagation delay this requirement is often not met. One way to circumvent this problem is to partition the channel capacity into a sufficient number of subchannels in time, so that the capacity of each subchannel is small enough that the outcome of one slot is known before the next slot could occur. In such partitioned channels, same CRA could independently be executed in each of the subchannel. This channelization, however, makes way for all those disadvantages of fixed assignment schemes, discussed in previous section, to crop up. Alternative approaches such as parallel tree search [28] or modified window protocol [27] have been proposed but, by and large, it is very difficult to achieve any substantial decrease in the irreducible propagation time-related delay.

Announced Random Retransmission Access (ARRA) [30] protocols proposed by Raychaudhuri achieves a higher throughput than the tree algorithm based CRA's. In addition, the average delay at low utilization levels is small, even on channels with high propagation delay. On balance, the ARRA techniques do not have unconditional stability which is characteristics of tree algorithm based CRAs. ARRA's was shown to give a capacity as high as 0.6. The principal mechanism by which the capacity improvement is achieved is same as that of tree based CRA: the collision between the new messages and retransmissions are prevented. ARRA uses a separate low capacity announcement subchannel to exchange control information to achieve it. In passing, we remark that it is not appropriate to relegate the ARRA into the combined random/reservation category because they neither require the maintenance of a global reservation queue nor involve elaborate initialization and recovery procedures which are characteristics of schemes that have a reservation component.

In URN scheme [20], users try to maximize the probability of successful packet transmission in each slot by choosing the optimum number of users permitted to transmit, using the estimate of the number of active users which have packets to transmit. This scheme behaves like the dynamic tree algorithm when ther are small number of users. As the number of users increases, however, its performance approaches that of the slotted ALOHA scheme. Furthermore, under long propagation delay, a significant performance degradation may arise if traffic fluctuation is large.

There also exists a class of Carrier Sense Multiple Access (CSMA) protocols which are very efficient for short propagation delay environments: short in comparison with packet transmission time. In CSMA the collision probability is significantly reduced by listening to the channel before transmitting: the packet is transmitted only if the channel is idle. CSMA protocols cannot be utilized for satellite systems because of its long propagation delay; for this reason CSMA is ubiquitous only in terrestrial networks.

The random access schemes exhibits the following characteristics:

- They are capable of supporting a large number of bursty users: their performance is insensitive to user population size and are therefore flexible with respect to future system growth.
- They have a low mean packet delay at low traffic load but the maximum channel utilization is generally much smaller than one.
- It is often difficult to provide fairness and priority handling capability.
- Packets may not be delivered in the correct sequence.
- The protocols are 'soft' and hence require the users to reliably detect collisions. They are sensitive to traffic load and have to exercise some control to be stable.

2.3 Demand Assignment (Reservation) Schemes

We have discussed two extremes in channel time allocation as far as control over the user's access right is concerned: the *hard* fixed assignment schemes which have the most rigid control, are nonadaptive to dynamically varying demand, and are thus wasteful in-terms of capacity utilization if small delay constraints is to be met; and *soft* random access schemes which involves less control, are adaptive to varying demands, but whose maximum channel utilization is highly constrained due to collisions. In this section we examine demand assignment techniques that exchange explicit information on the need for communication resource; and allocate channel resources according to one's demand [38]. These schemes are suitable when users generate infrequent bulks of data.

The active stations which have a message to transmit, first transmits a (channel) reservation-request, requesting a certain number of slots. Then channel request is honored by granting the requested channel time in two ways:

Centralized Control: A central *slot scheduler* collects the channel requests and allocates channel time (slots) to requesting stations. The scheduler can be placed either at one of the earth station (called the master control station) or on board the satellite. While the latter requires on-board processing capability for satellite. The former does not require on-board processing but the performance is much inferior; because when the scheduler is placed at the master control station, the time it takes for a reservation request to be honored is at-least two round trip delays as opposed to one for on-board controller case.

Distributed Control: By virtue of broadcast nature of the channel, all the users learn the channel requests of every other user and maintain a global reservation request queue. Then, all the users execute an identical algorithm on this global information to determine who should access the channel next. The performance is same as that of the on-board placed scheduler case, but the scheme is not as robust. Inconsistency in global information as seen by different dispersed users can result when an user wrongly detects a reservation request and, this could in the worst case, be catastrophic. Hence a certain control is indispensable for stable operation.

Since the channel is the only means of communication among the terminals, the reservation request should somehow be conveyed to the central scheduler (in centralized scheme), or to the other users (in distributed scheme). This can be achieved by dividing the channel into two distinct sub-channels: data sub-channel for data packet transmission and reservation sub-channel for reservation request transmission. These distinct sub-channels can be derived either in time or in frequency. The presence of reservation sub-channel presents a certain channel overhead. The slots in the data channel are assigned according to the requests received through the reservation channel. The problem of multi-accessing remains in the reservation channel. Either a fixed assignment scheme or a random access schemes can be used for reservation transmission. The idea of shifting the multiple access problem from the data channel to reservation channel is that it is possible to reduce the capacity wastage due to idle users (in fixed assignment scheme) or packet collisions (in a random access scheme) since the size of a reservation packet is substantially smaller than that of a data packet. The use of fixed assignment scheme versus random access scheme for reservation transmission involves a number of performance trade-offs. Similar factors that determine the choice between fixed assignment schemes and random access schemes for (data) packet transmission also apply here.

From previous discussion it is evident that when the users are bursty and large in number, it is advantageous to share the reservation channel using S-ALOHA, such a scheme is called ALOHA-Reservation. On the other hand, when the number of users is small TDMA access is used; called TDMA-Reservation. These schemes derive the reservation channel in time, in contrast, SRMA [35] derives in frequency. Reservation could also be piggybacked on the header of data packets. This method of requesting reservation presents the minimum overhead, as no separate slots and associated guard space are required. However piggybacking is possible only after a data slot is available to a station: how to assign data slots (initially to offer a chance for the station to piggyback and thereafter keep its option to further piggyback) is the vital problem in devising this method.

There also exists circuit oriented systems which are efficient if the channel holding time required is long in comparison with the set-up times required in allocating subchannels. The data channel can be frequency derived or time derived. The famous SPADE system [16], for example, has a pool of FDMA subchannels which get allocated on request. It uses one subchannel operated in a TDMA fashion with one slot per frame permanently allocated to each user to handle the requests and releases of FDMA circuits. Intelsat's MAT-1 system uses the TDMA approach. TDMA subchannels are periodically reallocated to meet varying needs of earth stations.

The demand access schemes exhibits the following characteristics

- When realized in a distributed fashion, it is not rugged. On the other hand centralized control requires OBP capability
- Fairness and priority can be easily incorporated
- They often require elaborate initialization and recovery procedures
- The minimum achievable delay is never less than two round trip delays
- the maximum channel utilization is less than one when a separate reservation subchannel is used.

2.4 Combined Random/Reservation Protocols

Random access schemes like S-Aloha are very efficient at low utilization while demand assignment schemes such as Aloha-Reservation exhibit better delay characteristics at mid and high utilization ranges. Therefore the idea arose of blending these two schemes in such a fashion as to achieve a performance which behave like random access at low utilization and like reservation access at high utilization. This idea has promoted a class of multiple access protocols called combined random/reservation access. The combined random/reservation concept have received immense attention [39] [5] [24].

Here too the reservation requesting could be done in a number of ways. In this respect the discussion presented in the previous section is directly applicable. SRUC uses a TDMA reservation channel: at the beginning of every data slots are a few reservation minislots fixed assigned to users for TDMA access. It operates as S-ALOHA as long as no collision occurs. when collision is detected, the system's protocol switches to a reservation protocol with a TDMA reservation channel. When the queue of reservation is cleared, it switches back to S-ALOHA. Thus, the system is always stable. Because of this adaptive property, SRUC exhibits good performance at low to high utilization ranges. Wieselthier and Ephremides also have studied similar schemes (named IFFO) that employs TDMA reservation along with random data packet transmissions [38].

When fixed assigned reservation channel is used, overhead grows in proportion to the population size. This is because reservation minislots have considerable amount of guard time associated with them, and number of minislots grows as population size is increased. Moreover, even with a significantly high overhead allocated for reservation subchannel, the fixed allocation makes a ready user to wait long for his reservation slot; inadaptability to dynamic activity is *per se* of fixed assignment schemes. For very large user population the S-ALOHA access to reservation is a natural choice: It inherits the adaptability, as well as insensitivity to number of users. Lee and Mark have studied this technique elegantly called by them as CRRMA scheme [23]. In CRRMA, when a station generate a new packet it is transmitted in the data portion (if the data portion has not been already reserved for a particular user) and simultaneously a reservation packet is also transmitted in one of the reservation minislot If the random data transmission succeeds then the reservation packet is neglected, otherwise the reservation will be stored at the On Board Scheduler and will be assigned slots. If both reservation as well as the data packet collide, the reservation packet alone is retransmitted; the data packet will be transmitted using the subsequently reserved data slots. Based on the acctions taken when the reservation request collide two algorithms were proposed: (i)Uncontrolled Channel Access (UCA): the collided reservations are retransmitted in one of the minislots randomly; new reservation packets could also contend (ii)Controlled Channel Access (CCA): the collided reservation requests alone are permitted to contend. When the users generate single packet messages with a Poisson arrival process, CCA requires at-least three minislots per data slot for the reservation scheme to be stable, while UCA requires at-least five minislots. UCA is simpler to implement but is inferior in terms of performance than CCA.

In Lee's CRRMA there seems to be one unnecessary restriction: the packet that arrives such a way that the upcoming slot is unreserved alone is allowed to take part in random transmission. If the upcoming slot is a reserved one it is fated, for no reason, to go through reserve - wait - transmit sequence. Possibly, it would have been conducive to achieving a lower delay if the packet waited until it saw a unreserved slot and then attempt ALOHA access. Controlled Multiple Access (CMA) [39] was devised to exploit this idea. CMA strives to achieve a well calculated balance between the volumes of packets transmitted immediately and those which make reservation. In CMA a number of slots are organized into frames and each frame is divided into two portions: reserved subframe and ALOHA subframe. In reserved subframe are the data slots that are dedicated to users in honor of their previously placed successful reservation requests; in ALOHA subframes are the data slots open for contention. Every slot also contains a number of reservation minislots associated. To ensure optimum channel performance under all traffic loading conditions, a control is placed on the relative rates of data traffic that is allowed to contend in ALOHA subframe and those that is relegated to make reservation only (and thereafter get transmitted). The optimum values of control parameter for minimum average packet delay depends on the channel load. The stations estimate the load and adjusts the value of control parameter accordingly. It is also shown that the scheme is not very sensitive to estimation errors.

In CREIR [25], along with random and explicit reservation, implicit reservation is incorporated. The idea of introducing implicit reservation is to support stream traffic in addition to bursty traffic. The bursty traffic is handled similar to Lee's CRRMA; for stream traffic, the first packet of the stream that gets through (either via explicit reservation or random access) has a 'one bit' in its header. This is an indication to all other users that this particular slot in every frame will be exclusively used by the stream. The last packet of the stream would have the header bit turned zero, relinquishing the exclusive right of the slot thereafter. CRIER is efficient in supporting both bursty as well as stream traffic.

2.5 Combined Fixed/Demand Assignment Protocols

When the user population size is small, the fixed assignment scheme are irrevocably preferred; when the user population is large the combined random/reservation schemes give good delay response. On the other hand, when the size of the user population is neither too large nor too small, the combined fixed/demand assignment category can perform well. Moreover these schemes do not involve random access thereby gets rid of the notorious collision detection problem. The idea of these schemes is to blend TDMA and reservation scheme in a fashion to achieve the best features of both. In these schemes every frame is divided into fixed assignment area and demand assignment area. Every user station has a dedicated slot in fixed assignment area in every recurring frame which serves the purpose of keeping down the delay at low utilization range. while the demand assignment area, adaptively changing its size according to load, serves to reduce the queuing delay predominant at high utilization range.

In RMA (Reservation Multiple Access) scheme [3] time is divided into frames, each frame consisting of reservation minislots followed by fixed assigned data slots and then reservation access data slots. Each user has a reservation minislot and a fixed assigned slot in every frame. In this scheme every user transmits all the packets queued at the beginning of the frame before the next frame begins. Each user transmits a reservation for number of packets queued at the instant or reservation minislot minus one: minus one because each user would get a fixed assigned slot in the current frame. Its reservation will be honored in the reservation access portion of the frame. The number of users is such that duration of the fixed assigned area of the frame is longer than the propagation delay; hence by the end of fixed assigned slots everybody would learn every other users' request and. create and access the reservation slots cooperatively.

In [2] Ahmadi has studied a protocol belonging to this category. Unlike RMA, the demand assigned slots are interleaved between the fixed assigned slots in a pre determined fashion. The amount of demand assigned slots interleaved between fixed assigned slots depends on the estimated channel traffic load. The stations access right to demand assigned slots is coordinated by exchanging reservation information through piggybacking on data packet transmitted in fixed assigned slots. This scheme does not require a minimum limit on the number of users; but require estimating load and may not be as adaptive to changing traffic conditions as RMA.

2.6 Combined Free/Demand Assignment Protocols

This is a new category proposed in this research [26]. It performs eminently for the same traffic environment for which combined fixed/demand assignment schemes perform well: when the user population is neither too large nor too small. Here, whenever there is demand the demanding stations are *demand* assigned slots, whenever there is no demand the slots are *freely* assigned to a certain station based on some heuristic information. Protocols of this category is the subject of this thesis and is studied elaborately in the forthcoming chapters.

Chapter 3

The Combined Free/Demand Assignment Multiple Access (CFDAMA) Protocols

3.1 Introduction

A need exists to organize the increasing number of information processing terminals into a network to exchange information and share processing capability. Geosynchronous satellite provides a benign broadcast/multiple access medium to realize this network. A major problem in the design of such packet satellite networks is to share the channel efficiently. A multiple access protocol is an algorithm to achieve this efficient sharing.

When propagation delay is very small the class of carrier sense multiple access (CSMA) schemes are suitable. For large propagation delay environments, like satellite communications, a number schemes such as TDMA, FDMA, pure random access, pure reservation access, hybrid random/reservation access, etc. are employed. The fixed assignment schemes like TDMA and FDMA allocate channel capacity to a station regardless of the station's dynamic activity and are thus wasteful in terms of resource utilization (except when the user population is very small). To overcome this inefficiency reservation access schemes have been proposed. But pure reservation access schemes suffer from long delays even at low channel utilization. To circumvent long delays at low utilization range, random access is incorporated with reservation. The random access and reservation access are blend in such a fashion as to achieve a performance which behave like random access at low utilization and like reservation access at high utilization. The combined random/reservation concept have received immense attention [5],[23],[39]. Another way to circumvent longer delays is to combine reservation with fixed assignment [2],[3]. In these schemes every frame is divided into fixed assignment area and demand assignment area. Every user station has a slot in fixed assignment area and it could request more slots in demand assignment area. The fixed assignment area serves the purpose of keeping down the delay at low utilization range while the demand assignment area serves to reduce the queuing delay predominant at high utilization range.

What we propose in this thesis is a class of protocols in which reservation is combined with free assignment to achieve short delays. The proposed schemes are different from the combined fixed/demand assignment schemes in that the users are not assigned slots at certain fixed positions in each frame. Whenever a slot is not reserved it is *freely* assigned to one of the users based on some heuristic information.

The suitability of a particular scheme for a given system depends on the system traffic model (such as population size, message length, message interarrival distribution, etc.) and performance requirements(such as propagation delay, delay, response, fairness, simplicity etc). For wideband packet satellite networks, considering Delay Vs Utilization as the figure of merit and confining to the homogeneous buffered user population generating single packet messages, the channel propagation delay normalized by the transmission time of a message is the key parameter characterizing the distributed communication problem. The relation between round trip delay and NT (where N is the user population size and T is the length of one data slot in seconds) can be used as a criterion for choosing a protocol for the given system environment[1] If $NT <<$ round trip propagation delay, the simple fixed assignment scheme such as FDMA or TDMA is suitable. If $NT >>$ round trip propagation delay, the class of Combined Random/Reservation schemes such as CRRMA [23], CMA [39] is suitable. On the other hand, when N is in the range comparable to round trip propagation delay(as is the case in wideband packet satellite networks, for a range of user population) neither of the above two schemes are singularly superior. It is in this environment the proposed protocol performs eminently.

3.2 The CFDAMA Protocols

The idea is to combine demand assignment with free assignment in channel time allocation. Whenever channel time is not reserved, it is freely assigned to one of the stations.

In these schemes, the demand assigned slots gets interleaved with free assigned slots in such a manner as to adapt to individual station's instantaneous traffic fluctuations, there by alleviating the queuing delay predominant especially at higher utilization range. As elaborated in chapter 2, the combined fixed/demand assignment schemes also strive to achieve the same effect. But CFDAMA achieves it more efficiently because of the following reason: Unlike the combined fixed/demand assignment schemes, no data slots are assigned to stations in fixed positions of the frame, therefore the scheme more closely adapts to instantaneous traffic fluctuations and hence a better delay response is expected. Besides efficiency , CFDAMA also inherits operational simplicity: it neither requires monitoring the channel load and adjust the frame length as in [2] nor does it require the number of users to be greater than a minimum as in [3].

In CFDAMA, the scheduling of data slots to user stations can be realized in a centralized fashion or in a distributed fashion. In centralized scheme, the access right

[1] strictly true only when the user population is homogeneous and generate data messages

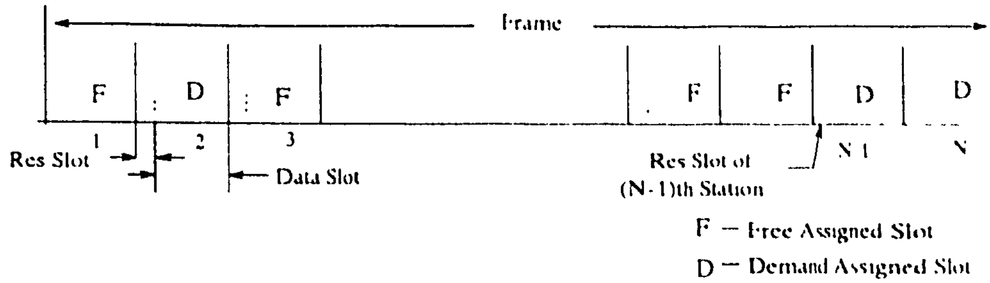


Figure 3.1: Sructure of the frame

of each slot is relayed to all stations by the *scheduler* placed on board the satellite. The scheduler could also be placed in one of the user stations, but the performance will suffer because it will take at least two round-trip-delays for a station to get its reservation request honored. In distributed scheme, each user executes an identical access algorithm to determine the access right of each upcoming slot. Distributed scheme does not require any on-board processing.

The reservation request can be made in a number of ways: using a separate reservation channel in which each user has a dedicated slot, using a separate reservation channel which is shared by all users on contention basis, and piggybacking the reservation request on the data packets. We examine CFDAMA using each of the above reservation methods in the following:

3.2.1 Fixed Assigned reservation (CFDAMA-FA)

In the following we explain the centralized scheme (with scheduler on board the satellite): The channel time is divided into contiguous slots. Each slot has a data portion(called Data Slot) long enough to hold a data packet and a reservation portion(called Res Slot) long enough to hold a reservation packet, shown in Figure 3.1. The channel is shared by N homogeneous buffered users. Each user has a Res Slot with a period of N slots. Therefore N slots are said to constitute a frame.

Each user station transmits reservations in its Res Slot for all new packets[2]

[2]'new packets' are those that arrived between the previous Res Slot and the present Res Slot

on behalf of which reservation has not been made and that remain at the instant of Res Slot. The On Board Scheduler(OBS) will place all the reservations in the reservation queue. It will serve one reservation every slot by *demand* assigning the Data Slot to the reserving station. Whenever the reservation queue becomes empty, the scheduler will *free* assign the upcoming Data Slot to one of the stations in a round robin fashion. The free assigned slot will be utilized if the assigned station happens to have a packet at the time the free assigned slot becomes effective at the ground. By doing so a minimum mean packet delay of *round trip delay*$+\frac{NT}{2}$ is achieved, as against 2*round trip delay* $+\frac{NT}{2}$ for pure reservation (demand assignment) schemes. It is noteworthy that unlike other reservation schemes, a packet on behalf of which a reservation is made may get transmitted even before its reservation is honored, either via a free assigned slot or via a demand assigned slot reserved by some antecedent packet of the same station.

Reserving for all newly arrived packets remaining at the instant of Res Slot might put the reserver in undue advantage: a number of demand assigned slots could go unused. We modify the reservation strategy as follows: Each user station keeps count of number of reservations that are yet to be honored for him. This count is called due slot count. The count is incremented by the number of slots requested each time a reservation is made, and is decremented by one whenever the user receives a demand assigned slot. The OBS would say whether the assigned slot comes out of demand assignment or free assignment. The reservation is made only for number of slots equivalent to[3] (number of packets queued in the station - due slot count)$^+$. The simulation results show that this *controlled* reservation scheme, improves the delay performance. The improvement achieved is pronounced

---

[3]

$$(x)^+ = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases}$$

when the frame length NT is such that its integral multiple is slightly smaller than round trip delay. This is because at such NT the number of demand assigned slots going unused is maximum for simple reservation[4]. Also, it appears, intuitively, that this controlled reservation would serve to reduce the variance of packet delay because the free slots are distributed to users whose packets are expected to expected to experience long delays. On the other hand, when NT is such that its integral multiple is slightly larger than round trip delay, the controlled and simple reservation perform identically. The performance of controlled and simple reservation obtained through simulation is compared in Figure 3.2.

In a distributed scheme, the users maintain their own copy of global reservation queue and access the Data Slot according to the above procedure. Inconsistency in global information can result when a user wrongly detects a reservation and this could cause, in the worst case, catastrophic failure of the distributed system. Hence a certain control is indispensable for stable operation.

3.2.2 Piggy-Backed Reservation (CFDAMA-PB)

Using fixed assignment reservation requires a separate reservation channel which presents certain overhead and, hence a fixed reduction in maximum possible channel utilization. Another way the reservation can be made is by piggybacking on data packets. Such a reservation scheme renders full channel utilization and a shorter frame length. This scheme is discussed below.

The channel time is divided into contiguous slots, each long enough to hold a data packet. The OBS maintains a reservation queue and a table of of all users' ID The User IDs are arranged arbitrarily to begin with. We shall call this table as free-assignment-table. Whenever the reservation queue is empty the OBS schedules the upcoming slot to the user at the top of the table. Consequently the user's ID is

[4]simple reservation means reserving for all newly arrived packets remaining at the instant of Res Slot

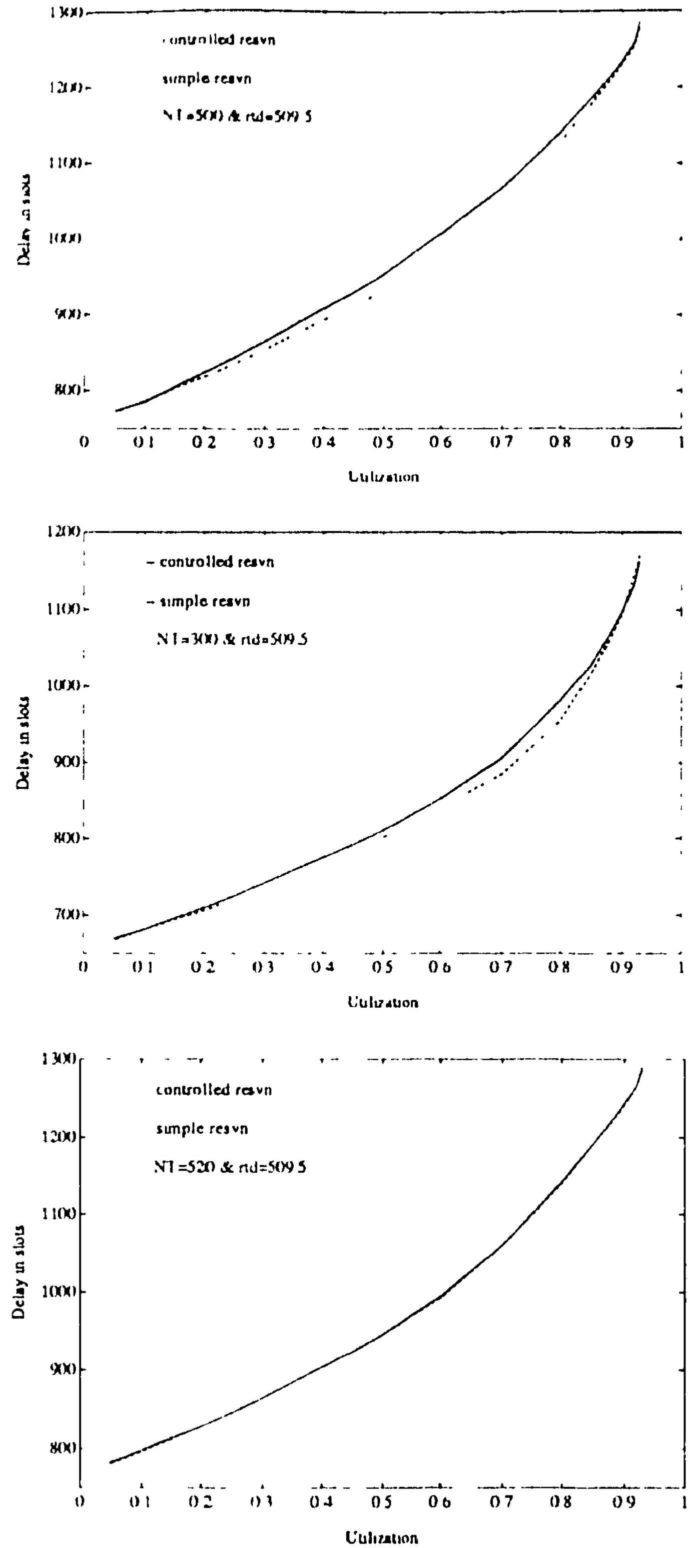


Figure 3.2: CFDAMA-FA with simple and controlled reservation

moved to the bottom of the table. Whenever the reservation queue is non empty, the slot is scheduled to the user at the head of the reservation queue. Users request for additional slots by piggybacked reservations by marking on the head of their data packets. The OBS places the received reservation requests in the reservation queue and serves them on FCFS basis. A straightforward reservation strategy would be to make reservation for all the new packets. 'New packets' are those on behalf of which a reservation has not been made as yet. However such a reservation strategy puts the reserving user in an undue advantage, costing the potential users waiting for a free assigned slot to wait longer. A more efficient reservation strategy is to adopt the *controlled* reservation as discussed before: reservations are piggybacked only for number of slots equivalent to (number of packets queued in the station − due slot count)$^+$.

At high channel utilization, most of the packets gets through via demand assigned slots. The busy period of the reservation queue server would be very long. Hence any user who does not have a reservation to be honored would feel neglected, waiting for a very long time for his free assigned slot. On the other hand, users who have reservations to be honored (and hence means to make further reservations) would still be getting free assigned slots occasionally. To correct this *unfairness*, as and when a user is scheduled a demand assigned slot, his ID is moved to the bottom of the table. We say, the user ID is reordered. By user ID reordering, we introduce a certain amount of priority over those stations that do not have a reservation to be honored, in allotting a free assigned slot. The performance improvement resulting from the controlled reservation strategy and user ID reordering obtained via simulation is shown in Figure 3.3. Some improvement in delay and a marked improvement in variance of delay is introduced, especially at high channel utilization range. Figure 3.4 elucidates the timing relations involved. Figure 3.5 summarizes the scheme in the form of a flow chart, when it is implemented as a distributed algorithm.

The simulation of CFDAMA-PB gives a delay response which compares favorably with the CFDAMA-FA, shown in Figure 3.6.

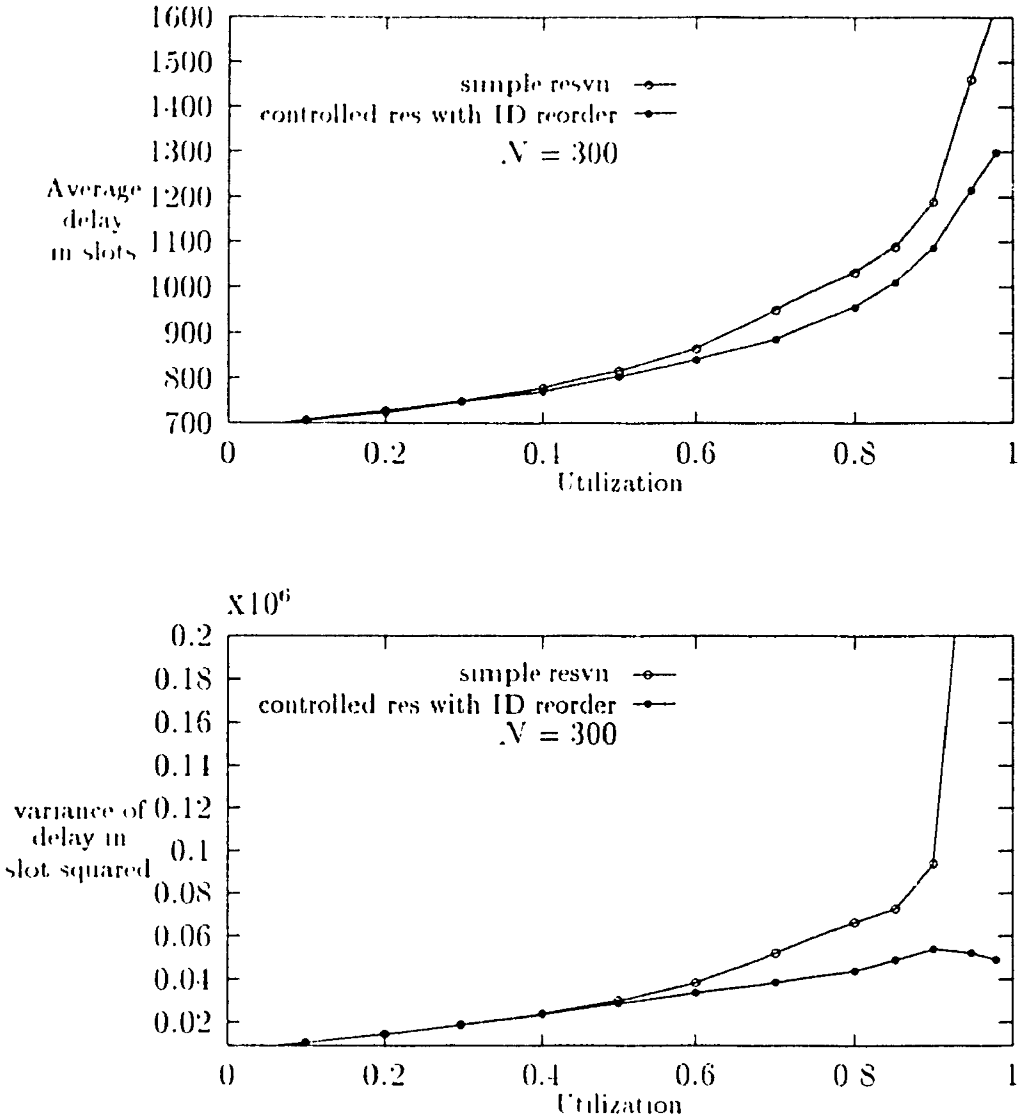


Figure 3.3: CFDAMA-PB with simple and controlled reservation with ID reordering

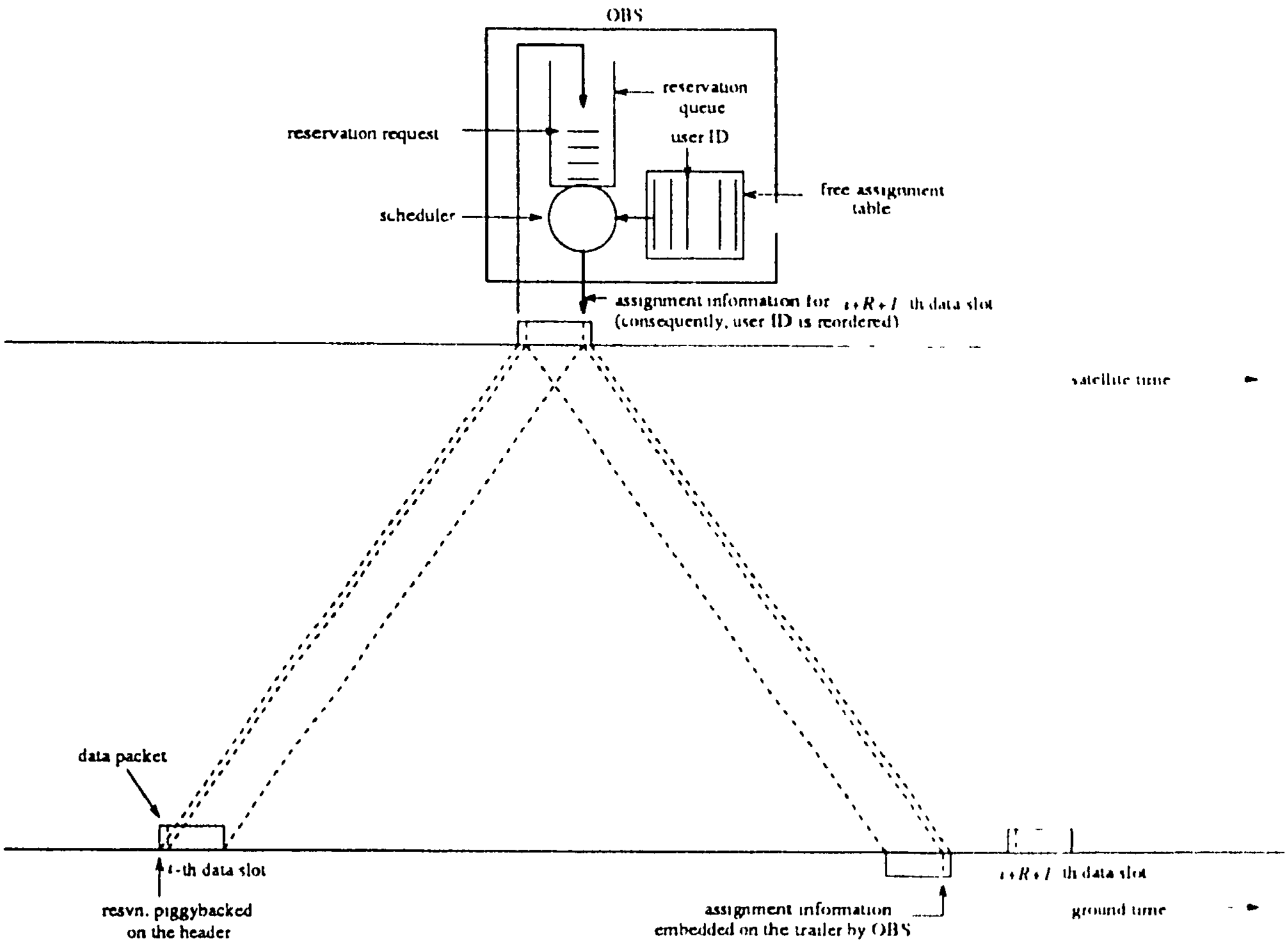


Figure 3.1: Slot Assignment by On Board Scheduler

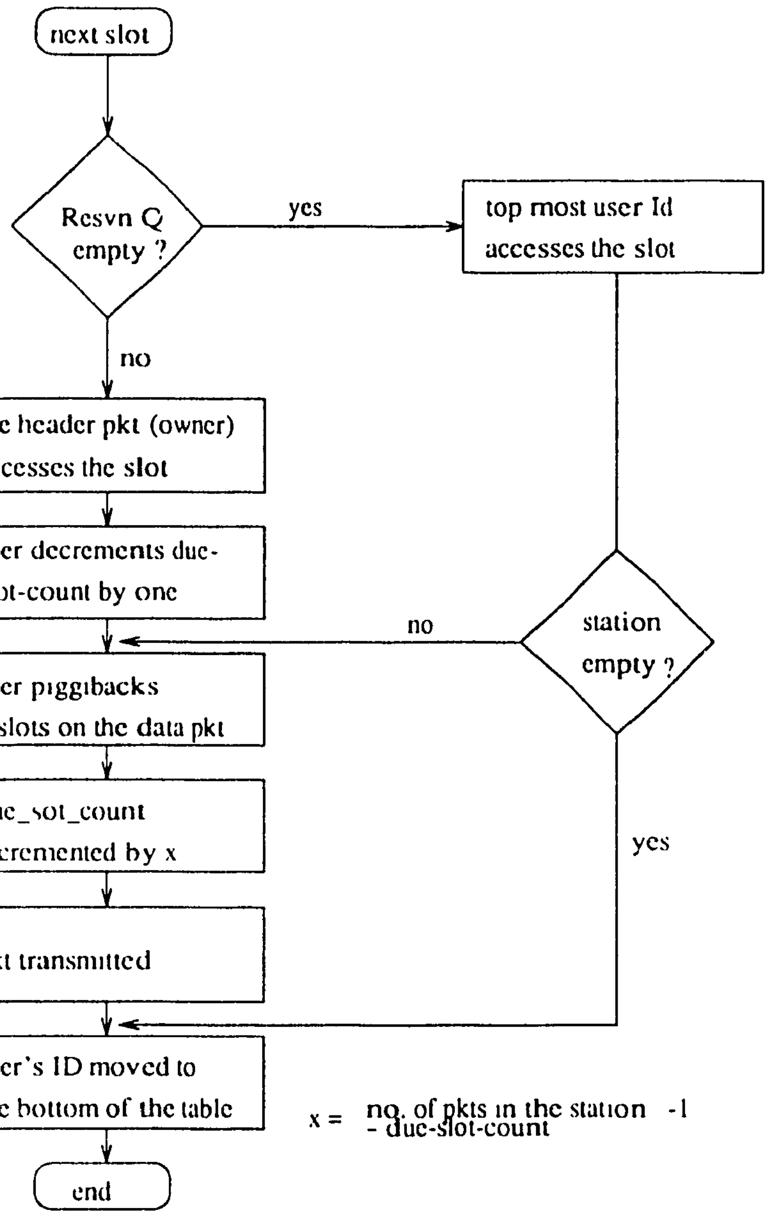


Figure 3.5: The accesss algorithm executed at the begining of every slot.

3.2.3 Random Access Reservation (CFDAMA-RA)

The third method of reservation is to allow ALOHA access to Res Slots. Unlike the previous methods this allows stations to make reservation whenever a packet is generated. Though this Aloha access has the advantage of easy incorporation of new users, our initial simulation study shows the scheme to be inferior as compared to the FA and PB versions, except when N is very large. when N is very large it would perform almost identically with pure Aloha Reservation scheme. Moreover, at such large N, the hybrid random/reservation schemes are unquestionably eminent. In our study, we used one Res Slot every Data Slot and circumvented the stability problem by just neglecting collided reservations. Those packets whose reservations collided will have to escape via free assigned slots. This strategy works well at low utilization range but pays off at high utilization range. The simulation results are shown in Figure 3.6.

Finally, it would be a good idea to allow random access and piggybacking to coexist. The piggybacking would serve the purpose of keeping the channel wastage due to reservation channel at the minimum and random access to Res Slots would provide the user with the flexibility of accessing it at any time.

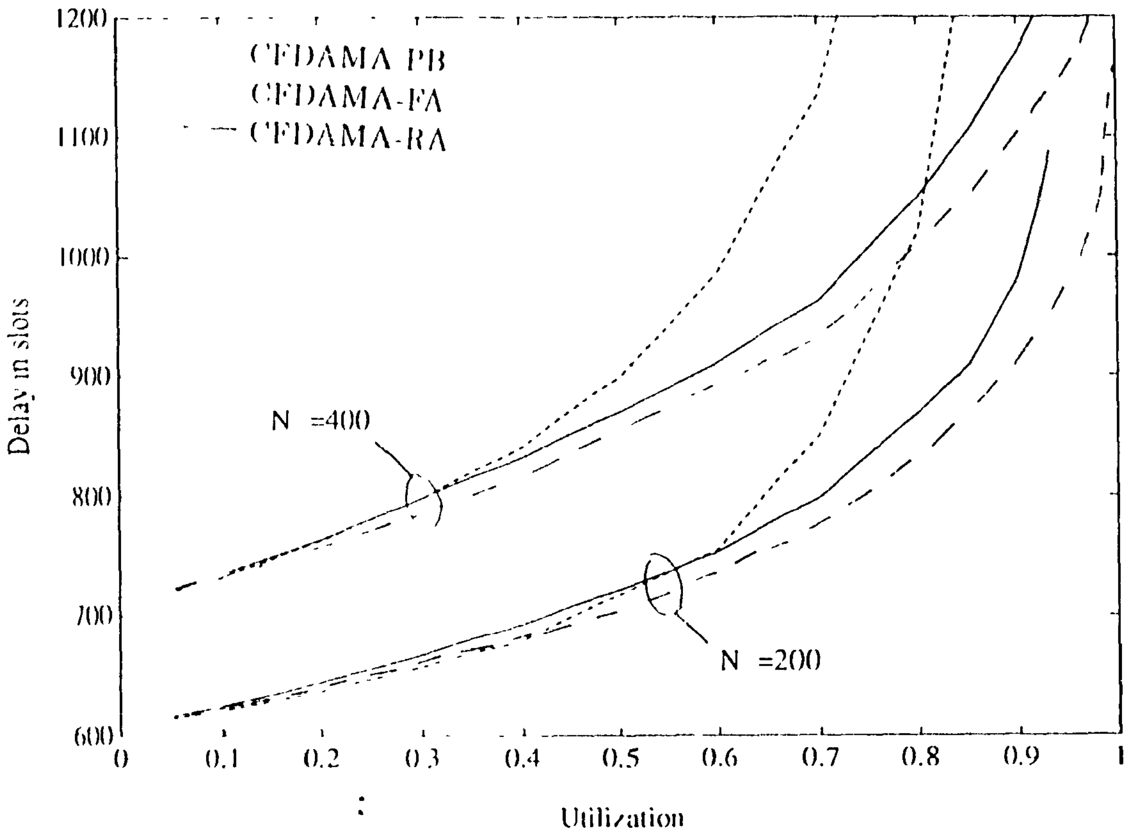


Figure 3.6: Comparison of different versions of CFDAMA

Chapter 4

Modeling and Performance Analysis

We consider the following traffic model:

- There are N homogeneous information processing terminals (user stations) each with unlimited buffer capacity.

- The arrival process at each user station obeys Poisson law with mean S/N i.e. the probability of l arrivals to an user station in a slot interval is

$$Pr(A = l) = \frac{e^{-S/N}(S/N)^l}{l!}$$

  where S represents the overall system input load. The Poisson assumption represents a worst case situation in the sense that the Poisson statistics corresponds to a large degree of randomness.

- The generated messages are of constant length and constitute a single packet. (Hence the terms packet and message are used interchangeably.)

We model and analyze FA and PB versions of CFDAMA within a queuing theoretic framework.

4.1 CFDAMA-FA

We concentrate on a tagged packet that arrives at a station. We follow its flow through the system until it reaches its destination and attempt to evaluate the delay it incurs, on the average, in its transit. Each successful packet is distinguished into three classes. Class1 packets are those that gets through via *free assigned slots.* Class2 packets are those that gets through via *undue reserved slots.* (i.e., a packet succeeds using a slot that has been reserved by some antecedent packet of the same station) and Class3 packets are those that gets through via *due reserved slots.* (i.e., a packet reserves and uses the honored slot for its transit). A packet's average delay depends on the class it assumes. Our analysis essentially reduces to evaluating the average delay of each class of message and the probability with which a tagged message assumes a particular class.

A packet can succeed as Class1 packet either before or after making a reservation on its behalf. The free assigned slots of a station is separated, on the average, by a distance of $\frac{N}{1-d}$. Where, N is the population size and d is the fraction of slots that are demand assigned. The exact distribution of the distance between the free assigned slots is characterized by the way in which the demand assigned slots gets interleaved between them and is very difficult to determine. A packet will assume Class1 status if it meets a free assigned slot before meeting a demand assigned slot. We assume that the packet in the head of the queue makes Bernoulli attempts to assume Class1 status. (The Bernoulli attempts inherits the memoryless property and is often employed [33], [19] to carry through an analysis.) The Bernoulli attempts continue until the packet escape either as Class1, Class2 or Class3 packet.

A packet will assume Class2 status if it meets an undue reserved slot before meeting a free assigned or due reserved slot. We consider a particular user station, say ith station, and concentrate at its Res-Slot instants. Those packets that arrived in the previous frames and remain in the station at the instant of Res-Slot are old packets and those that arrived in the current frame are new packets.(see Figure 4.1).

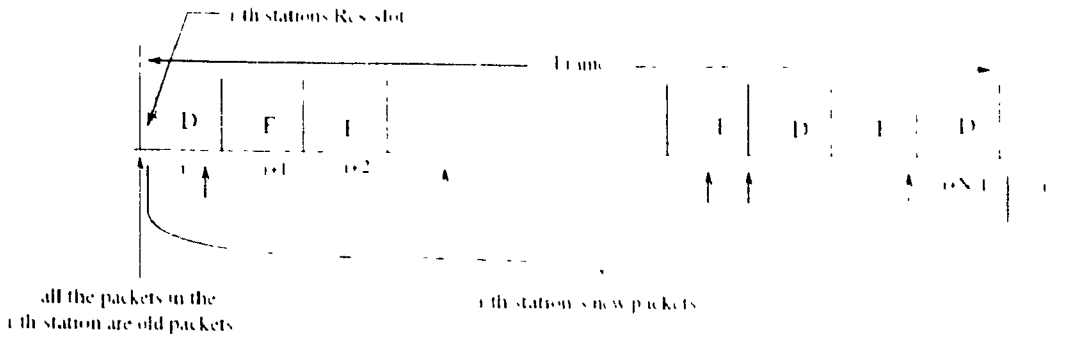


Figure 4.1 New and old arrivals of a station

According to the scheme, every old packet will have a demand assigned slot to its credit(to be honored). Whenever an old packet escapes as Class1, its due reserved slot will be grabbed by a new packet, if any. Our second crucial approximation is that a new packet can assume Class2 status if and only if there were at least one old packet at the beginning of the frame and an old packet escapes as Class1 packet.

A packet can remain in the station until it receives its due reserved slot, whereby it assumes Class3 status.

The average delay of the tagged packet is derived conditioning on the number of new arrivals in the current frame.

We define

$$S_l \triangleq \text{mean number of reservations in scheduler queue}$$

$$E(D_k|l,q) \triangleq \text{average delay of the } k\text{th new arrival in getting a slot given that there were } l \text{ new arrivals and q old packet}$$

we have

$$E(D_1|l, q=0) = \sum_{i=1}^{R+S_l+\frac{Nl}{l+1}+1} i p'(1-p')^{i-1}$$

$$+ (1 - \sum_{i=1}^{R+S_q+\frac{Nl}{l+1}+1} p'(1-p')^{i-1})(R+S_l+\frac{Nl}{l+1}+1) \tag{1.1}$$

where

$$p' = \frac{2(1-d)}{N} \tag{1.2}$$

The first term accounts for Class1 status while the second one accounts for Class3 status. No Class2 is possible when q=0 We do not use the exact distribution of the arrival instant of the kth packet but are satisfied with using its mean arrival instant. Since the l packets are uniformly distributed in the frame, the kth packet arrives on the average in $\frac{Nk}{l+1}$th slot and it would take $N - \frac{Nk}{l+1}$ slots before it could see its Res slot. The timing relations are illustrated in Figure 1.2. The computation of equation 1 is shown in the form of probability tree in Figure 1.3

$$E(D_2|l, q=0) = \sum_{i=1}^{R+S_l+\frac{Nl}{l+1}+1} p'(1-p')^{i-1}. \sum_{j=\max(i,\frac{N}{l+1})}^{R+S_q+\frac{Nl}{l+1}+2} p(1-p)^{j-\max(i,\frac{N}{l+1})}(j-\frac{N}{l+1}).$$

$$+ \sum_{i=1}^{R+S_l+\frac{Nl}{l+1}+1} p'(1-p')^{i-1}.(1 - \sum_{j=\max(i,\frac{N}{l+1})}^{R+S_l+\frac{Nl}{l+1}+2} p(1-p)^{j-\max(i,\frac{N}{l+1})}).(R - \frac{N(l-1)}{l+1} + S_q + 2)$$

$$+ (1 - \sum_{i=1}^{R+S_q+\frac{Nl}{l+1}+1} p'(1-p')^{i-1}).(R - \frac{N(l-1)}{l+1} + S_q + 2) \tag{1.3}$$

where

$$p = \frac{(1-d)}{N} \tag{1.4}$$

The first term accounts for both the first and the second packet assuming Class1 status, the second term accounts for first packet assuming Class1 status while the

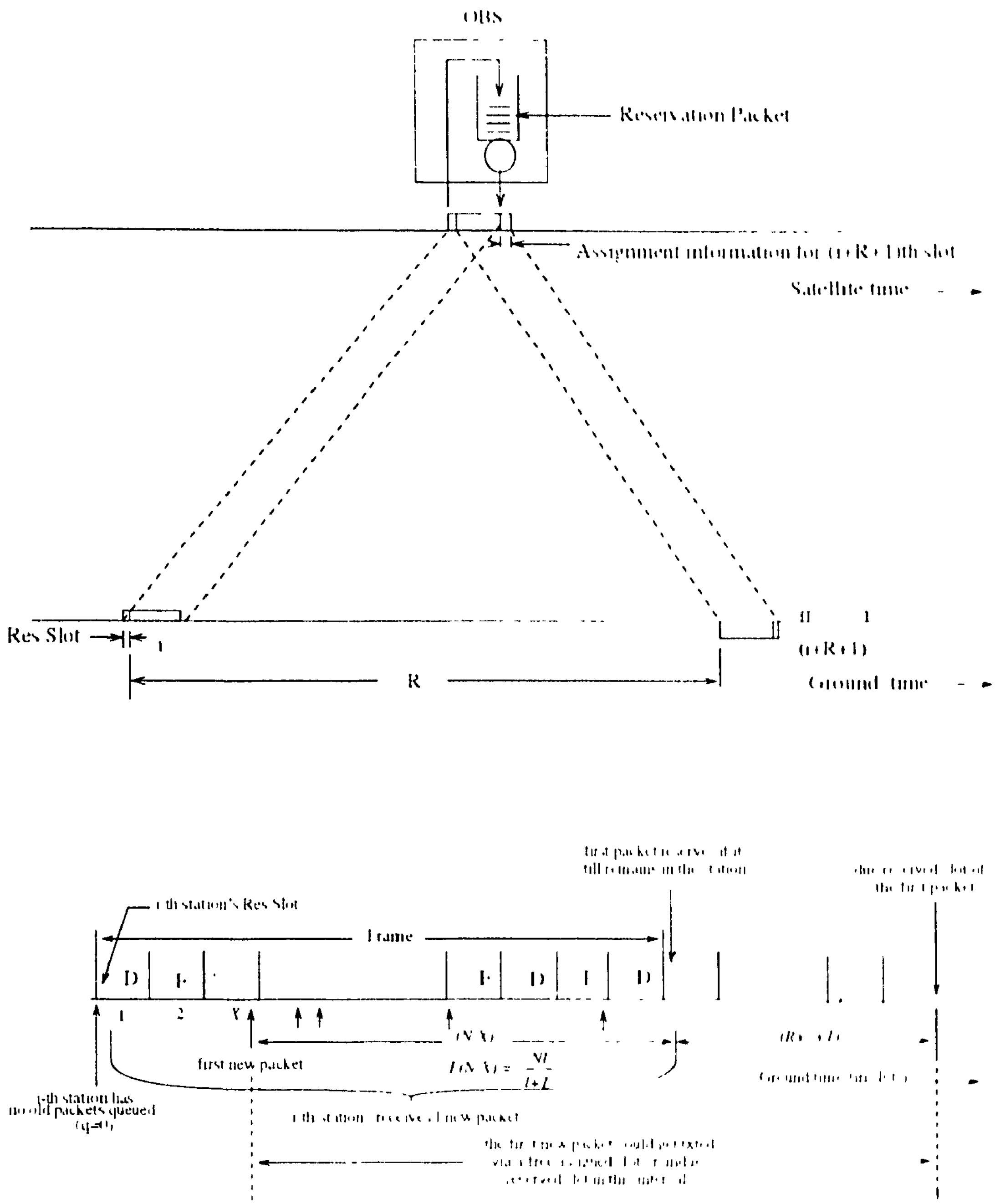


Figure 1.2: Illustration of timing relations.

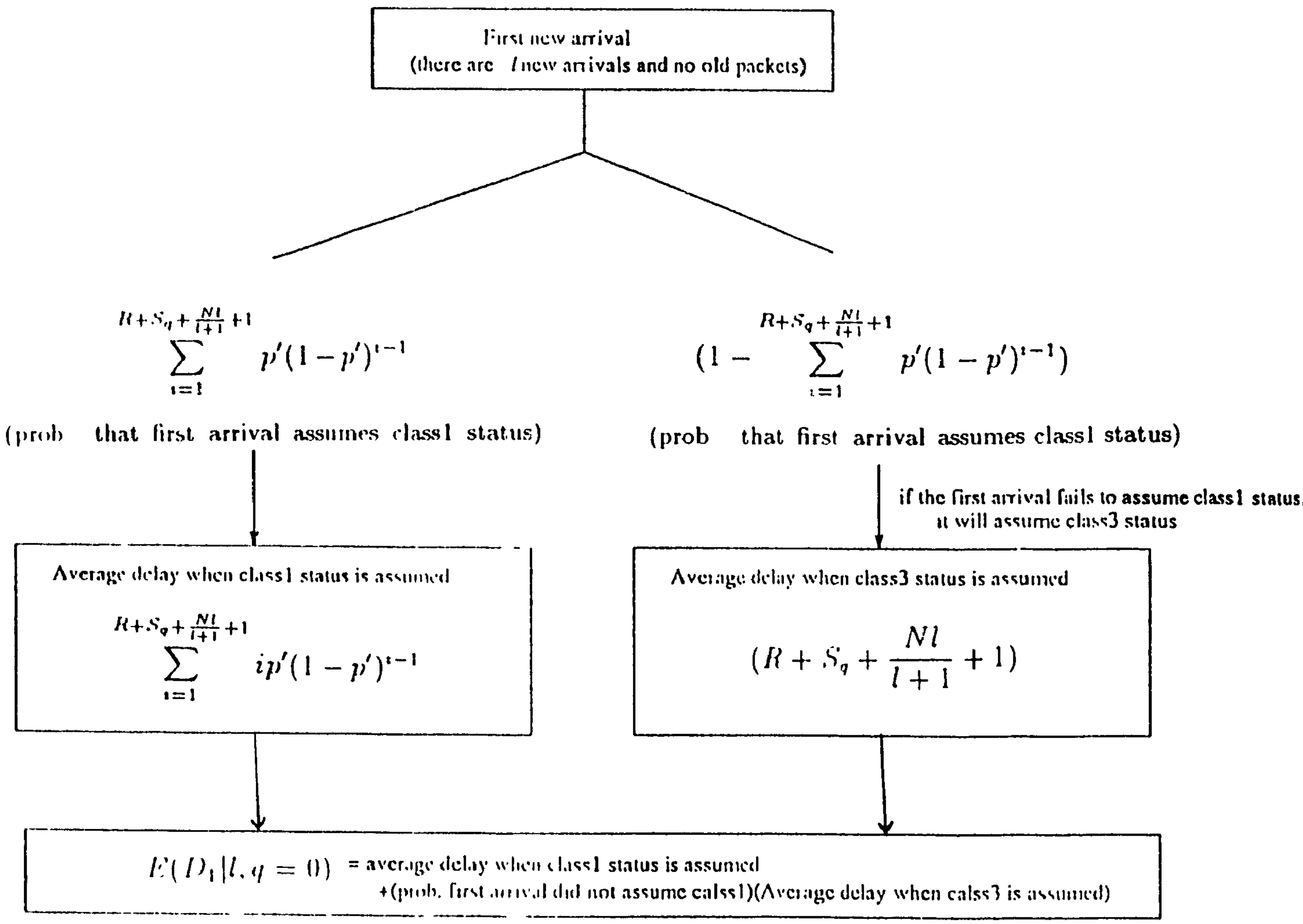


Figure 1.3 The probability tree for equation (1.1)

second assume Class3 status and the third term accounts for both the first as well as the second packet assuming Class3 status. writing expressions for the higher values of k follows the same approach but is highly cumbersome. We verified through computation that the values of these expressions are very closely approximated by

$$E(D_k|l,q=0,k\geq 2) = N - \frac{kN}{l+1} + R + S_q + k \tag{15}$$

Which is the average delay for a packet assuming Class3 status, an intuitively pleasing result.

Similarly, for the case $q>0$, we have

$$E(D_1|l,q=1) = \sum_{i=1}^{R+S_q+1} p'(1-p')^{i-1}.\{\sum_{j=\max(i,\frac{N}{l+1})}^{R+S_q+1} p(1-p)^{j-\max(i,\frac{N}{l+1})}(j-\frac{N}{l+1})$$

$$+(1-\sum_{j=\max(i,\frac{N}{l+1})}^{R+S_q+1} p(1-p)^{j-\max(i,\frac{N}{l+1})}).(R+S_q-\frac{N}{l+1}+1)\} + (1-\sum_{i=1}^{R+S_q+1} p'(1-p')^{i-1})$$

$$\{\sum_{j=R-\frac{N}{l+1}+S_q+1}^{R+S_q+\frac{Nl}{l+1}+1} jp(1-p)^{j-(R+S_q-\frac{N}{l+1}+1)} + (1-\sum_{j=R-\frac{N}{l+1}+S_q+1}^{R+S_q+\frac{Nl}{l+1}+1} p(1-p)^{j-(R+S_q-\frac{N}{l+1}+1)})$$

$$(R+\frac{Nl}{l+1}+S_q+1)\} \tag{16}$$

$$E(D_1|l,q>1) = \sum_{i=1}^{R+S_q+1} p'(1-p')^{i-1}(R+S_q-\frac{N}{l+1}+1) + (1-\sum_{i=1}^{R+S_q+1} p'(1-p')^{i-1})$$

$$\{\sum_{j=R-\frac{N}{l+1}+S_q+1}^{R+S_q+\frac{Nl}{l+1}+1} jp(1-p)^{j-(R+S_q-\frac{N}{l+1}+1)} + (1-\sum_{j=R-\frac{N}{l+1}+S_q+1}^{R+S_q+\frac{Nl}{l+1}+1} p(1-p)^{j-(R+S_q-\frac{N}{l+1}+1)})$$

$$(R+\frac{Nl}{l+1}+S_q+1)\} \tag{17}$$

and

$$E(D_k|l,q>0,k>1) = N - \frac{kN}{l+1} + R + S_q + k \tag{18}$$

Now unconditioning,

$$E(D|l) = \sum_{i=1}^{l} \sum_{q=0}^{\infty} P_q E(D_i|l,q) \tag{4.9}$$

$$E(D) = \frac{1}{1 - e^{-S}} \sum_{l=1}^{\infty} \frac{a_l}{l} E(D|l) \tag{4.10}$$

In which

$$a_l = \frac{e^{-S} S^l}{l!} \text{ and}$$

P_q = probability that q old packets are queued at the instant of Res Slot

After getting a slot, the packet will reach the destination after a round trip delay. Therefore,

$$\text{The average message delay} = E(D) + R$$

The unknown quantities in the above equations are d, S_l, P_0 and P_1. The fraction of slots that are demand assigned, d, is given by the average number of arrivals in a frame that make reservation. The expression for average number of arrivals that make reservation depends on the way in which N and R are related. When $N > R$, d is obtained as a solution to the equation

$$d = (1-d)\left(\sum_{l=1}^{\infty} x_l(l-1)a_l + (1-x_l)la_l\right) \tag{4.11}$$

$$+d\left(\sum_{l=1}^{\infty} y_l(l-1)a_l + (1-y_l)la_l\right) \tag{4.12}$$

where

$$x_l = \left\{\begin{array}{l}\text{prob. frame has a} \\ \text{free assigned slot}\end{array}\right\} \cdot \left\{\begin{array}{l}\text{prob. first arrival came before free assigned} \\ \text{slot given } l \text{ new arrivals}\end{array}\right\}$$

and

$$y_l = \left\{\begin{array}{l}\text{prob. frame has a} \\ \text{free assigned slot}\end{array}\right\} \cdot \left\{1 - \begin{array}{l}\text{prob. free assigned slot occurred} \\ \text{after } R\text{th slot and before the} \\ \text{first arrival, given } l \text{ new arrivals}\end{array}\right\}$$

The probability function of the arrival instant of the first packet conditioned that there were l new arrivals is given by

$$\Pr(I_{dt}) = Pr\left(\begin{array}{l}\text{first arrival occurs at } t\text{ th slot of the frame} \\ \text{given there were } l \text{ new arrivals}\end{array}\right)$$

$$= \{\prod_{i=1}^{t-1}(1 - \frac{l}{N-i+1})\}\frac{l}{N-t+1} \qquad (9.13)$$

In writing the above equation we have assumed that a user station does not generate more than one packet in one slot interval, which is highly reasonable because of the large N. Given the frame has a free assigned slot, its position is uniformly distributed within the frame. Now it is straightforward to show

$$x_l = (1-d)\sum_{t=1}^{N-l+1}\Pr(I_{dt})\frac{N-t}{N} \qquad (9.14)$$

and

$$y_l = (1-d)(1 - \sum_{t=R}^{N-l+1}\Pr(I_{dt})\frac{t-R}{N}) \qquad (9.15)$$

The evaluation of d for the case $N < R$ follows the same approach except that now we have to take into account the happenings in $\lceil\frac{R}{N}\rceil$ frames. Where $\lceil a\rceil$ denotes smallest integer greater than a.

The d evaluated as above agrees very well with simulation results. The comparison is given in Figure 9.1 for two values of N.

The arrival process at the scheduler queue consists of bulk arrivals at every Res. Slot. The bulk size depends on the number of new arrivals that will make reservation. The probability generating function of the bulk size for the case $N \geq R$ is given by

$$B(z) = (1-d)(\sum_{l=1}^{\infty} x_l z^{(l-1)}a_l + (1-x_l)z^l a_l) \qquad (9.16)$$

$$+d(\sum_{l=1}^{\infty} y_l z^{(l-1)}a_l + (1-y_l)z^l a_l) \qquad (9.17)$$

Knowing $B(z)$, the mean number of reservations in the scheduler queue is computed as

$$S_t = B'(1) + \frac{B''(1)}{2(1-B'(1))} \qquad (9.18)$$

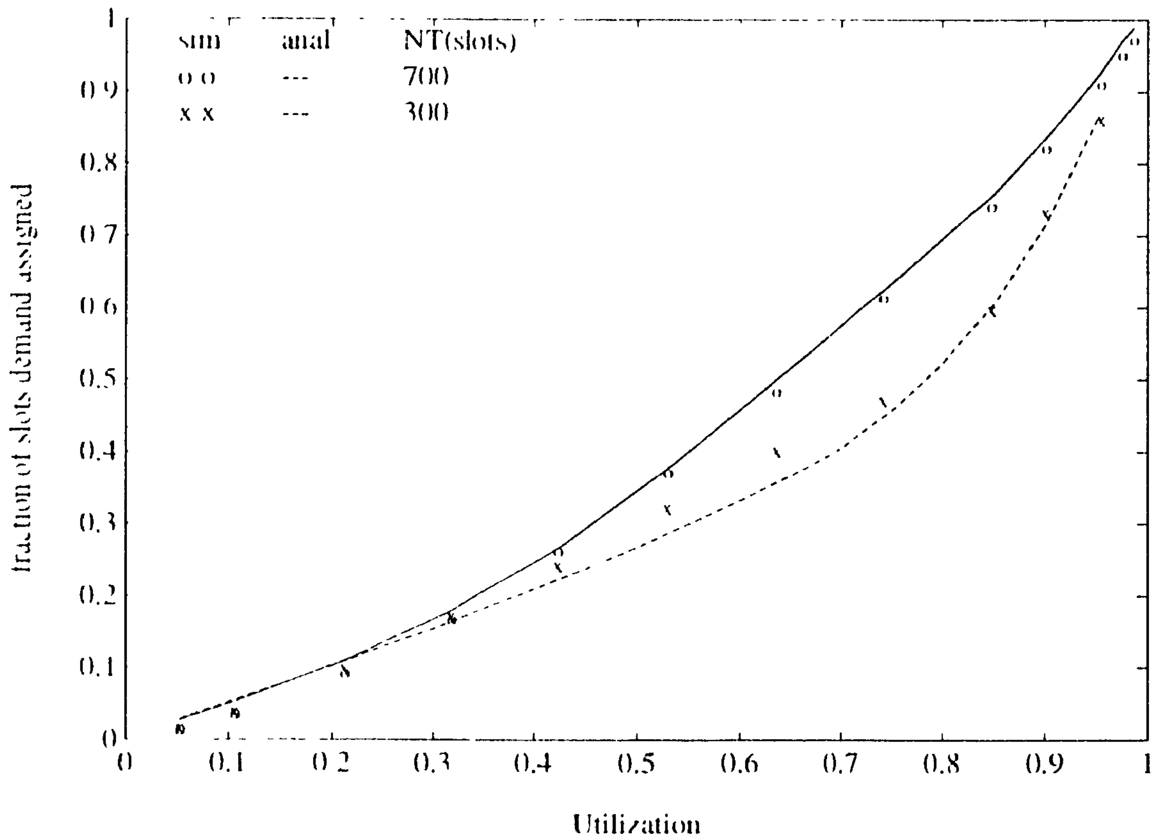


Figure 4.4: Comparison of analytic and simulation results

Finally, P_0 and P_1 are obtained by assuming that the user stations are served by continuously available server who take a geometrically distributed length of time to serve a packet. The mean service time of a packet arriving to a nonempty station is Λ slots and mean service time of a packet arriving to an empty station is $\frac{\Lambda}{2(1-\theta)}$ slots. We have,

$$P_0 = \frac{1 - S}{1 - S + \frac{S}{2(1-\theta)}} \tag{4.19}$$

$$P_1 = \frac{1 - \hat{G}(0)}{G(0)} P_0 \tag{4.20}$$

$G(z)$ and $\hat{G}(z)$ denotes the probability generating function of the number of packets that arrive during the service time of the packet which arrived to a nonempty station and an empty station respectively.

The performance of CFDAMA-FA obtained by simulation and analysis are compared in Figure 4.5.

4.2 CFDAMA-PB

The analysis of CFDAMA protocol that uses piggybacking as a means of making reservation is complicated by the fact that the interval between reservation instants[1] is quite *random* as is not the case with most of the multiple access schemes analyzed in the literature. Since reservation is piggybacked, the reservation instants of a station are the instants at which it receive data slots. In turn, the reception of data slots by a station depends also on the reservation it had piggybacked previously. This dependency makes the exact mathematical modeling of the scheme difficult.

To the author's knowledge, no scheme that employ piggybacking as a means of making reservation has been analyzed in the literature. In [2] an analysis is given for a scheme that employs piggybacking as a means of making reservation. But

[1]'reservation instants' are those at which an user can make reservation

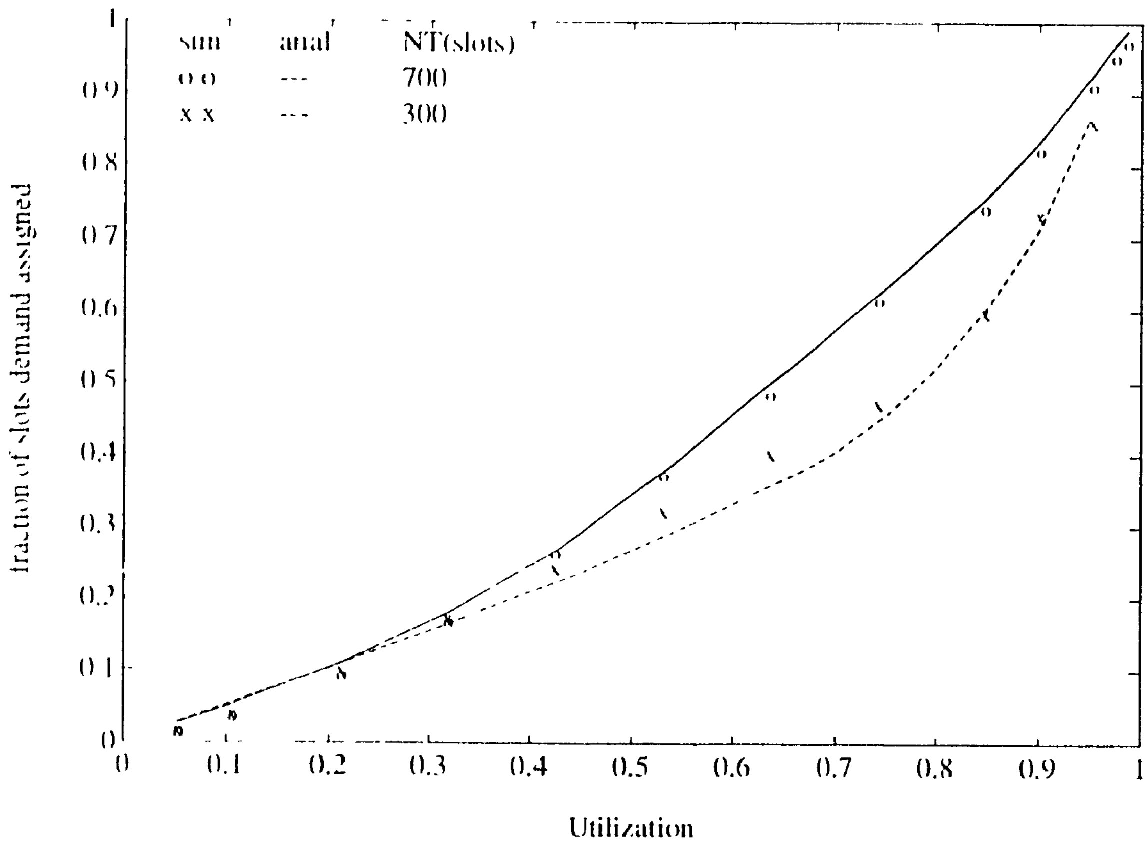


Figure 4.5: Comparison of analytic and simulation results

then, there piggybacking is *not pure* in the sense that the stations were not allowed to piggyback reservation through any data packet. A definite frame structure was introduced, where the users are provided with data slots at certain fixed positions of the frame and, the users where *restricted* to reserve only through these fixed assigned slots (and not through demand assigned slots). This restriction, at the cost of some performance degradation, makes the reservation instants to recur at *deterministic* intervals and hence allows the scheme to be modeled as a Markov Process. Here, we provide an approximate analysis of CFDAMA as it is, that is without imposing any such restriction or frame structure.

In this analysis, in addition to the adopted traffic model, we assume that the round trip propagation delay normalized by the transmission time of a packet (R) between any two user stations is smaller than the user population size (N).

We tag a random packet that arrives at a station, say m th station, and evaluate the delay it incurs until it reaches the destination. The availability of channel to the station[2] is modeled as an intermittently attending server. The server could arrive to the station in either of the following two modes

1. demand mode (i.e., the server arrives as a result of demand assignment, we designate the server as *called server*) or

2. free mode (i.e., the server arrives as a result of free assignment, we designate the server as *free server*).

The delay of the tagged packet depends on whether, on its arrival, the forthcoming server is a called server or a free server. Hence, we adopt the following steps to develop the pgf of packet delay: we obtain the probability with which either type of servers happen to be the forthcoming one, and then we obtain the conditional pgf of delay given that the forthcoming server is of a particular type, and finally we compute the unconditional pgf of delay.

---

[2]hereafter, 'station' refers to the particular m th station

As a first step, we obtain the probability with which either type of server happens to be the forthcoming one in the following: when the tagged packet arrives at the station, the forthcoming server can be a free server or a called server. The type of the forthcoming server depends on whether the station was empty or nonempty when the precedent server departed from the station. It will be a called server if the station was nonempty, else, it will be a free server. In order to obtain the status of the station (i.e., empty or nonempty) at server departure instants, we develop the pgf of the number of packets queued in the station at these instants. The basic equation governing the number of packets queued in the station at server departure instants is

$$n_{i+1} = U(n_i)a_i + [1 - U(n_i)]\tilde{a}_i \tag{4.21}$$

where,

$$U(n_i) = \begin{cases} 0 & \text{if } n_i = 0 \\ 1 & \text{if } n_i > 0 \end{cases}$$

$n_{i+1} \triangleq$ the number of data packets queued in the station at the $(i+1)$th server departure instant.

$a_i \triangleq$ the number of packets arriving between the ith and $(i+1)$th server departure instants, given that the ith server departure left the sation non-empty.

$\tilde{a}_i \triangleq$ (the number of packets arriving between the ith and $(i+1)$th server departure instants - $1)^+$, given that the ith server departure left the station empty.

The above equation embodies the following facts: Whenever a called server arrives, it will serve only those packets that were queued when the last server departed. Whenever a free server arrives it will serve at the most one packet that may arrive since the last server departure. Now, it is straight forward to derive the pgf of the number of packets queued in the station at the server departure instants to be[3]

$$P(z) = (1 - P_0)\, A(z) + P_0\, \hat{A}(z) \tag{4.22}$$

where,

$$\hat{A}(z) \triangleq \text{the pgf of } \hat{a}_i.$$

$$A(z) \triangleq \text{the pgf of } a_i.$$

$$P_0 \triangleq \text{the probability that the queue is empty.}$$

The expressions for $A(z)$ and $\hat{A}(z)$ is to be developed later. From the above equation the quantity of interest, P_0, is calculated as,

$$P_0 = \frac{A(0)}{1 + A(0) - \hat{A}(0)} \tag{4.23}$$

A well known attribute of the point process is that the distribution of the interdeparture interval in which the random packet (tagged packet) arrives and the distribution of the general interdeparture interval are not the same. The random packet is more likely to select longer intervals.

we define,

$l_j \triangleq$ the probability that the interval between the ith server and $i+1$th server departure instants is j slots, given that the $i+1$th server is a called server.

$\hat{l}_j \triangleq$ the probability that the interval between the ith server and $i+1$th server departure instants is j slots, given that the $i+1$th server is a free server.

$L(z) \triangleq$ the pgf of l_j.

$\hat{L}(z) \triangleq$ the pgf of $\hat{l}_j$.

[3]Obtaining $P'(1)$, and applying the Little's formula does not give the average delay of a packet because the state of the system at server departure instants is non typical

$E_f \triangleq$ the event that the tagged packet arrives at the station such that the forthcoming server is a free server.

$E_c \triangleq$ the event that the tagged packet arrives at the station such that the forthcoming server is a called server.

$\lambda \triangleq \frac{S}{N}$, the average number of arrivals per station per slot.

$d_n \triangleq$ the delay of the tagged packet in getting a server is n slots.

Now, from the theory of point process [7], we have the desired expressions,

$$\Pr(E_f) = \frac{P_0\tilde{L}'(1)}{P_0\tilde{L}'(1) + (1 - P_0)L'(1)} \tag{4.24}$$

and

$$\Pr(E_c) = \frac{(1 - P_0)L'(1)}{P_0\tilde{L}'(1) + (1 - P_0)L'(1)} \tag{4.25}$$

Now as a second step, we derive the conditional pgf of delay given that the forthcoming server is of particular type. We consider the free server type first.

$$\Pr(\text{tagged packet arrives in an isdi}^4 \text{ of } j \text{ slots } |E_f) = \frac{j\tilde{l}_j}{\tilde{L}'(1)} \tag{4.26}$$

Given that the tagged packet has arrived in an inter server departure interval of j slots, its arrival is uniformly distributed in the interval $(1, j)$ with probability function $\frac{1}{j}$. If the tagged packet happens to be the first packet to arrive in the

[4] inter server departure interval is abbreviated as isdi.

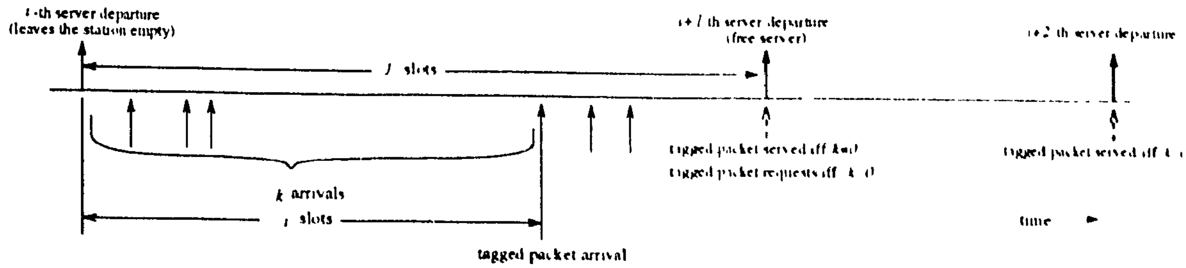


Figure 4.6: Tagged packet arrival and its departure

interval, it will be served right away by the forthcoming free server. Otherwise, it will have to wait for an additional interval distributed according to $L(z)$, to be served by the next server. It is so because, every time the free server comes to the station it can serve only one packet. Also, the next server will be a called server since the tagged packet will make a request, in case it was not served by the forthcoming one. The timing relations involved are illustrated in Figure 4.6.

The probability that the tagged packet has to wait for n slots to get served is given by the following sum:

$$
\begin{aligned}
\Pr(d_n|E_f) \;=\;& \Pr\begin{pmatrix}\text{tagged pkt selects}\\ \text{an isdi of } j \text{ slots}\end{pmatrix}.\Pr\begin{pmatrix}\text{it arrives at the}\\ (j-n)\text{th slot}\end{pmatrix}.\\
& \Pr\begin{pmatrix}\text{no arrivals in}\\ (j-n)\text{ slots}\end{pmatrix}\\
+\;& \Pr\begin{pmatrix}\text{tagged pkt selects}\\ \text{an isdi of } j \text{ slots}\end{pmatrix}.\Pr\begin{pmatrix}\text{it arrives at}\\ (j+i-n)\text{th slot}\end{pmatrix}\\
& \Pr\begin{pmatrix}\text{there is atleast 1}\\ \text{arrival in } (j+i-n)\text{ slots}\end{pmatrix}\Pr\begin{pmatrix}\text{the next isdi}\\ \text{is of } i \text{ slots}\end{pmatrix}
\end{aligned}
$$

The first term in the summation accounts for the tagged packet being served by a free server and the second term accounts for the tagged packet being served by a called server. From Figure 4.6, the pgf of the number of slots the tagged packet should wait until it gets a server follows as

$$D(z|E_s) = \sum_{j=R}^{\infty} \frac{jl_j}{L'(1)} \sum_{i=1}^{j} \frac{1}{j} \cdots z^{\cdots} + \sum_{k=1}^{\infty} \frac{(\lambda(j-i))^k}{k!} e^{-\lambda(j-i)} z^{i} L(z)) \quad (4.27)$$

Now we derive the conditional pgf of the delay of the tagged packet given that it arrives such a way that the forthcoming server is a called server. Under this condition the tagged packet cannot be served by the forthcoming server, it has to wait for the subsequent called server[3]. The probability that the tagged packet has to wait for n slots to get served is given by

$$\Pr(d = n|E_c) = \Pr\begin{pmatrix}\text{tagged pkt selects}\\ \text{an isdi of } j \text{ slots}\end{pmatrix} . \Pr\begin{pmatrix}\text{it arrives at}\\ (j+i-n)\text{th slot}\end{pmatrix} . \Pr\begin{pmatrix}\text{the next isdi}\\ \text{is of } i \text{ slots}\end{pmatrix}$$

and the pgf is

$$D(z|E_c) = \sum_{j=R}^{\infty} \frac{jl_j}{L'(1)} \sum_{i=1}^{j} \frac{1}{j} \sum_{k=0}^{\infty} \frac{(\lambda(j-i))^k}{k!} e^{-\lambda(j-i)} z^i L(z) \quad (4.28)$$

In writing the above equations, we assume that the tagged packet reserves through the last packet served by the forth coming server and it is served last by the subsequent called server. This does not cause any inaccuracy since k is very much smaller than the inter server departure interval (i.e. the probability that k assume values compatable to interserver departure interwal is very negligible.

Finally, the unconditional pgf of the delay of the packet, in getting a slot, is

$$D(z) = D(z|E_c)\Pr(E_c) + D(z|E_f)\Pr(E_f) \quad (4.29)$$

The pgf of the packet delay in reaching the destination is $D(z)z^{R-1}$. The term z^{R-1} accounts the time the packet takes to reach the destination from the instant it gets under service. The unknown quantities in computing the above equations are

[3]the subsequent server will be a called server for sure, since the tagged packet will make a request through a packet that will be served by the forthcoming server

$A(z)$, $\hat{A}(z)$, $I(z)$ and $\hat{I}(z)$. We have,

$$A(z) = I(e^{\lambda(1-z)}) \qquad (4.30)$$

$$\hat{A}(z) = \sum_{n=N}^{\infty} I_n (e^{-\lambda} + \lambda n e^{-\lambda} + \sum_{j=2}^{\infty} [illegible] \frac{(\lambda n)^j e^{-\lambda}}{j!}) \qquad (4.31)$$

$$= \hat{I}(e^{-\lambda})(1 - z^{-1}) + (1 - \lambda) \frac{d}{d\lambda} I(e^{-\lambda}) + z^{-1} I(e^{\lambda(1-z)}) \qquad (4.32)$$

The above equations follows the fact that whenever a called server arrives, it will not serve any arrivals since the last server departure instant and whenever a free server arrives it will serve off the first arrival since the last server departure. The key quantities to be computed are $I(z)$ and $\hat{I}(z)$. We compute $I(z)$ first.

A called server arrives when a packet of the station makes a request. The request goes to the OBS(onboard scheduler) and possibly experience some delay at the scheduler queue and then cause the scheduler to release a called server. Hence if $S(z)$ is the delay encountered by the request in the scheduler queue, then

$$I(z) = S(z) z^{R} \qquad (4.33)$$

In the above equation, we omit the number of slots the server would stay at a station as this would be a negligibly small quantity yet make the equations cumbersome. The requests arrive at scheduler queue in bulks with a highly complex interarrival distribution. An approximate $S(z)$ is obtained by estimating the actual average arrival rate of requests to the scheduler queue and assume them to arrive according to Poisson distribution. The average arrival rate of requests at the scheduler queue λ_s is estimated as follows,

$$\lambda_s = N\lambda [\Pr(E_c).\Pr(\text{the tagged packet reserve } E_c) + \Pr(E_f).\Pr(\text{the tagged packet reserve } E_f)] \qquad (4.34)$$

in which,

$$\Pr(\text{the tagged packet reserve } E_c) = 1 \qquad (4.35)$$

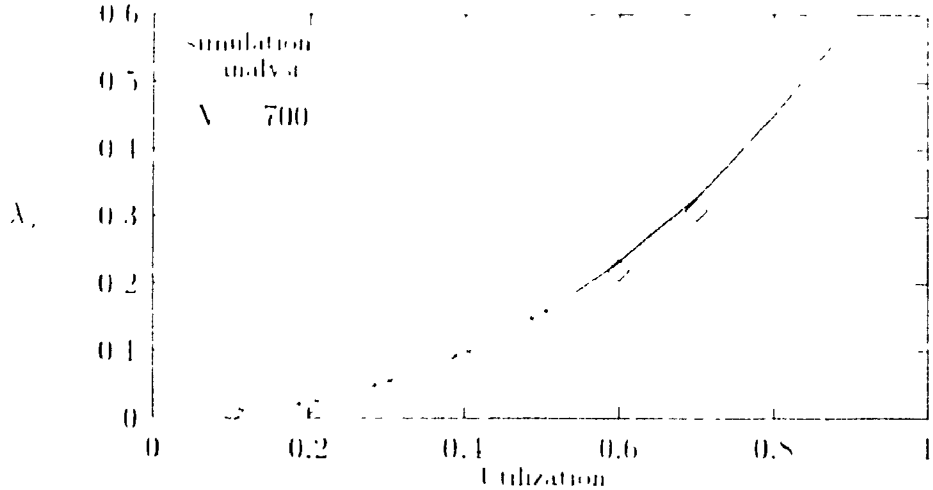


Figure 4.7: Comparison of estimated and simulation results

$$\Pr(\text{the tagged packet reserve}|E_f) = \sum_{j=N}^{\infty} \frac{j\tilde{l}_j}{\tilde{L}'(1)} \sum_{i=1}^{j} \frac{1}{j}(1-e^{-\lambda(j-i)}) \quad (4.36)$$

$$= 1 - \frac{1-\tilde{L}(e^{-\lambda})}{\tilde{L}'(1)(1-e^{-\lambda})} \quad (4.37)$$

The λ_s evaluated as above was compared with the one obtained by simulation. It was found to be in good agreement (shown in Figure 4.7). Now, with the assumptions made, it is simple to show

$$S(z) = \frac{(1-\lambda_s)(1-z)A_s(z)}{A_s(z)-z} \quad (4.38)$$

where,

$$A_s(z) = e^{-\lambda_s(1-z)} \quad (4.39)$$

Next we develop $\tilde{I}(z)$. A free server arrives if the station was empty when the last sever departed. The time it takes for the free server to arrive from the time the last server departed is given by the time it takes for the User ID to move from the tail of the free-assignment-table to the head[6]. The way the User ID advances to the top of the table is quite complicated. The scheduler will release a free server

[6]Besides the performance improvement, the user ID reordering makes the mathematical modeling and analysis particularly tractable.

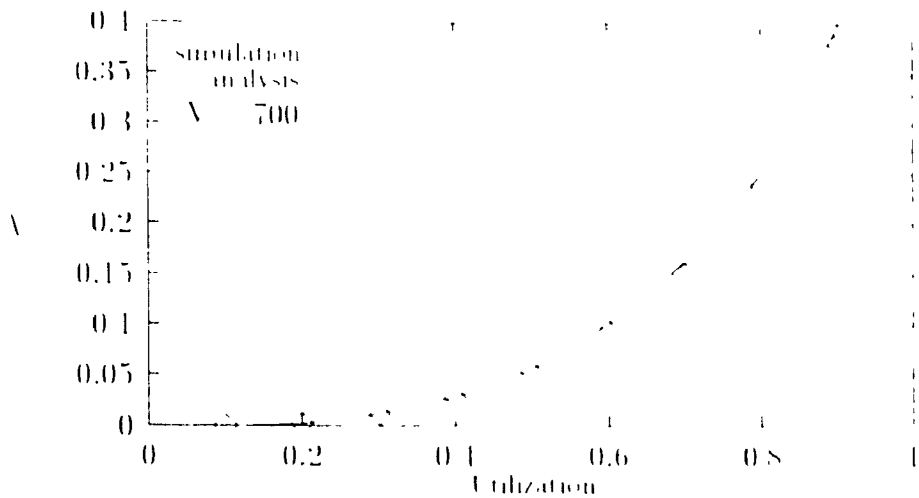


Figure 4.8: Comparison of estimated and simulation results

to the user at the top of the free-assignment table whenever the scheduler queue is empty, and will release a called server whenever the scheduler queue is not empty. According to the scheme, the tagged User ID will move up by one position whenever the scheduler releases a free server. On the other hand, when the scheduler releases a called server it may or may not move the User ID up. If the tagged User ID occupies a lower position than the User ID of the user to whom the called server is being released, the User ID will move, otherwise not. Thus not all called servers (in other words not all requests) effect advancement of tagged User ID.

Getting the exact $\tilde{L}(z)$ is extremely difficult owing to the complexity involved in exactly describing the advancement process of User ID. We adopt the following technique to compute an approximate $\tilde{L}(z)$: We estimate the average arrival rate of requests to the scheduler queue that *effectively hinder* the tagged User ID's advancement and assume the arrival to be a Poisson process. A procedure to estimate this arrival rate, denoted by λ is given in the appendix. The estimated values were found to be in very good agreement with the values obtained by simulation (shown in Figure 4.8). The User ID will move up whenever the queue receiving packets at a rate of λ is empty. During the busy period (BP) the advancement is completely hindered and during the idle period the User ID is moved one position up every slot time. Now, the pgf of the number of slots required to move the User ID from the bottom of the free assignment table to the top is computed as follows:

$$l_n = \binom{N}{i} . \Pr\left(\begin{array}{c} i \text{ BPs were} \\ \text{initiated} \end{array}\right) \Pr\left(\begin{array}{c} \text{all the } i \text{ BPs together} \\ \text{lasted for } n-i \text{ slots} \end{array}\right) \quad (4.10)$$

$$= \binom{N}{i} [(e^{-\lambda})^{N-i}(1-e^{-\lambda})^i].\{b^i(n-i)\} \quad (4.11)$$

Taking z transform on either sides, we have the desired pgf.

$$L(z) = \sum_{i=0}^{N} \binom{N}{i} (ze^{-\lambda})^{N-i}(z(1-e^{-\lambda})\hat{B}(z))^i \quad (4.12)$$

$$= (ze^{-\lambda} + z(1-e^{-\lambda})\hat{B}(z))^N \quad (4.13)$$

where

$\hat{B}(z)$ = the pgf of the busy period of the queue.(the ˆ is used to emphasise that the arrival rate to the queue is $\hat{\lambda}$ and not λ.)

An implicit equation for the pgf of the busy period can be obtained by identifying sub busy periods within the busy period, all of which have the same distribution as the busy period itself. The equation is developed below:

$$\Pr(BP = t) = \Pr\left(BP = t \,\middle|\, \begin{array}{c} \text{a bulk of n pkts} \\ \text{initiate the BP} \end{array} , \begin{array}{c} \text{j of the n slots} \\ \text{have bulk arrivals} \end{array}\right) . \quad (4.14)$$

$$\Pr\left(\begin{array}{c} \text{a bulk of n pkts} \\ \text{initiate the BP} \end{array}\right) . \Pr\left(\begin{array}{c} \text{j of the n slots} \\ \text{have bulk arrivals} \end{array}\right)$$

$$b(t) = \sum_{n=1}^{t} \frac{\lambda^n e^{-\lambda}}{n!(1-e^{-\lambda})} \sum_{j=0}^{n} \binom{n}{j} (1-e^{-\lambda})^j e^{-\lambda(n-j)} b^j(t-n) \quad (4.15)$$

$$\hat{B}(z) = \sum_{i=1}^{\infty} z^{i} \sum_{n=1}^{\infty} \frac{\lambda^{n} e^{-\lambda}}{n!(1-e^{-\lambda})} \sum_{j=0} \binom{n}{j} (1 - e^{-\lambda})^{j} e^{-\lambda(n-j)} \ldots$$

$$= \frac{e^{-\lambda}}{1-e^{-\lambda}} (e^{\lambda z[(1-e^{-\lambda})\hat{B}(z)+e^{-\lambda}]} - 1) \qquad (4.17)$$

It is possible to obtain a numerical solution for $\hat{B}(z)$ at any value of z through the following iterative equation:

$$\hat{B}_{n+1}(z) = \frac{e^{-\lambda}}{1-e^{-\lambda}} (e^{\lambda z[(1-e^{-\lambda})\hat{B}_n(z)+e^{-\lambda}]} - 1) \qquad (4.18)$$

in which we choose $\hat{B}_0(z) = 0$; for $\lambda < 1$, the limit of this iteration scheme will converge to $\hat{B}(z)$ [18].

To see how accurate is our developed $\hat{L}(z)$, we obtained the first and second moments of $\hat{L}(z)$ through simulation and compared with those obatined from the above developed equation. They were found to be in perfect agreement shown in Figure 4.9.

Now, we obtain explicit expressions for the average packet delay and the variance of packet delay

$$\text{Average Packet Delay}, D_{\text{ave}} = D'(1|E_s)\Pr(E_s) + D'(1|E_f)\Pr(E_f) + R + 1 \qquad (4.19)$$

$$\text{Variance of Packet Delay}, D_{\text{var}} = [D''(1|E_s) + D'(1|E_s)(1 - D'(1|E_s))]\Pr(E_s)$$
$$+[D''(1|E_f) + D'(1|E_f)(1 - D'(1|E_f))]\Pr(E_f)$$

where,

$$\Pr(E_f) = \frac{P_0 L'(1)}{P_0 L'(1) + (1 - P_0)\hat{L}'(1)}$$

$$\Pr(E_s) = 1 - \Pr(E_f)$$

$$D'(1|E_f) = \frac{1}{\hat{L}'(1)} \sum_{j=N}^{\infty} \sum_{i=1}^{j} (\hat{l}'(1)(1 - e^{-\lambda}) \ldots)$$

$$= \frac{1}{\hat{L}'(1)} (\frac{\hat{L}''(1)}{2} + \hat{L}'(1)(1 + \hat{L}'(1)) - \hat{L}'(1)(\frac{1}{1-e^{-\lambda}} \ldots$$

$$D''(1,F_a) = \frac{1}{L'(1)}\left(L'''(1)\frac{7-2\lambda}{6} + L''(1)(1-2\lambda+(\lambda+1)L'(1))\right.$$

$$+L'(1)\left(\frac{11-10\lambda}{6} + 2L'(1)(1+e^{-\lambda}) + L''(1)\right)$$

$$\left. L''(1)\left(\frac{e^{\lambda}(L(e^{-\lambda})-1)}{1-e^{\lambda}}\right) - \frac{2e^{\lambda}L'(1)(L(e^{-\lambda})-1)}{1-e^{\lambda}}\right)$$

$$D'(1,F_a) = \frac{L''(1)}{2L'(1)} + L'(1) + 1$$

$$D''(1,F_a) = \frac{1}{L'(1)}\left(L''(1)L'(1)(2+\lambda) + 2L'^2(1) + \lambda^2\frac{L''(1)-L'(1)}{6} + \right.$$

$$\left.\lambda(L'''(1)+4L''(1)+2L'(1)) + (L'''(1)+6L''(1)+5L'(1))\frac{1-2\lambda}{6}\right) - \frac{L''(1)}{2}$$

The average packet delay and the variance of packet delay obtained by analysis and simulation are compared in Figure 4.10 and are found to be in good agreement.

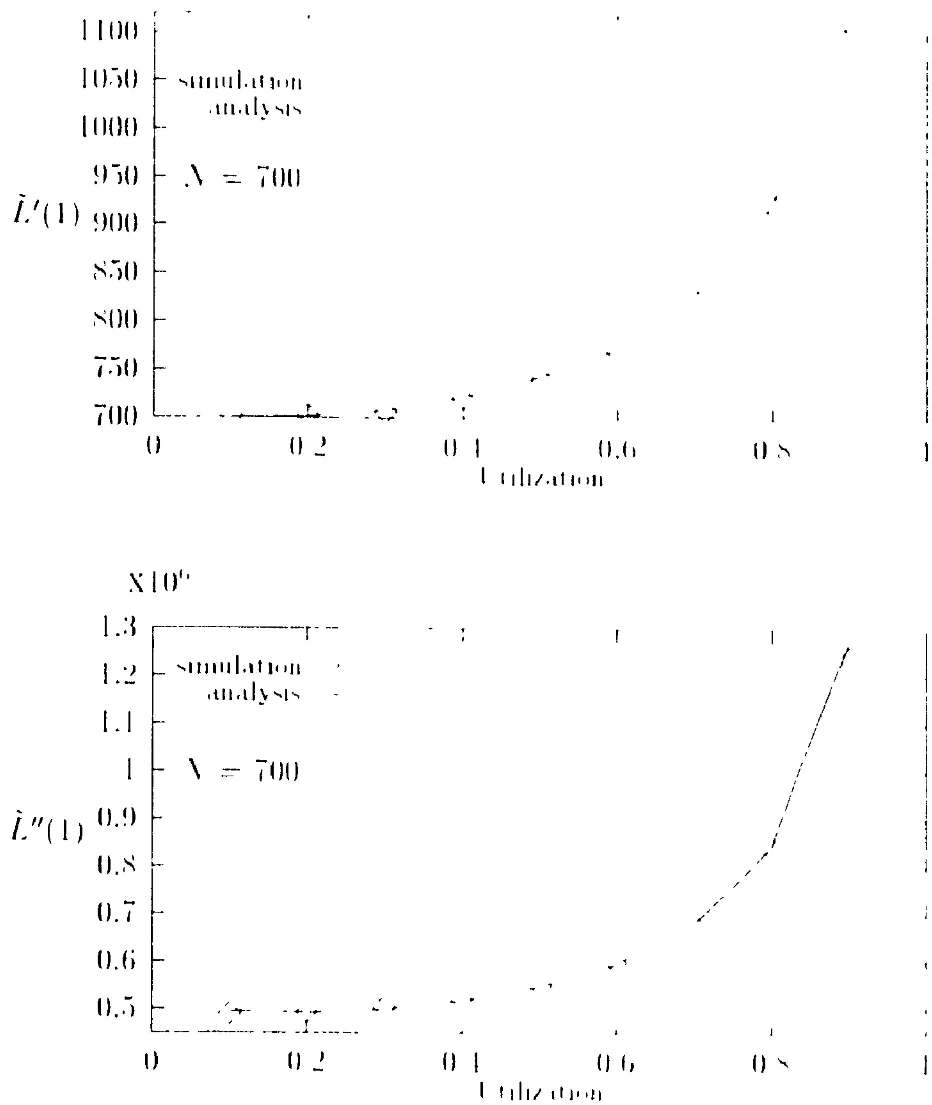


Figure 4.9: Comparison of analytic and simulation results

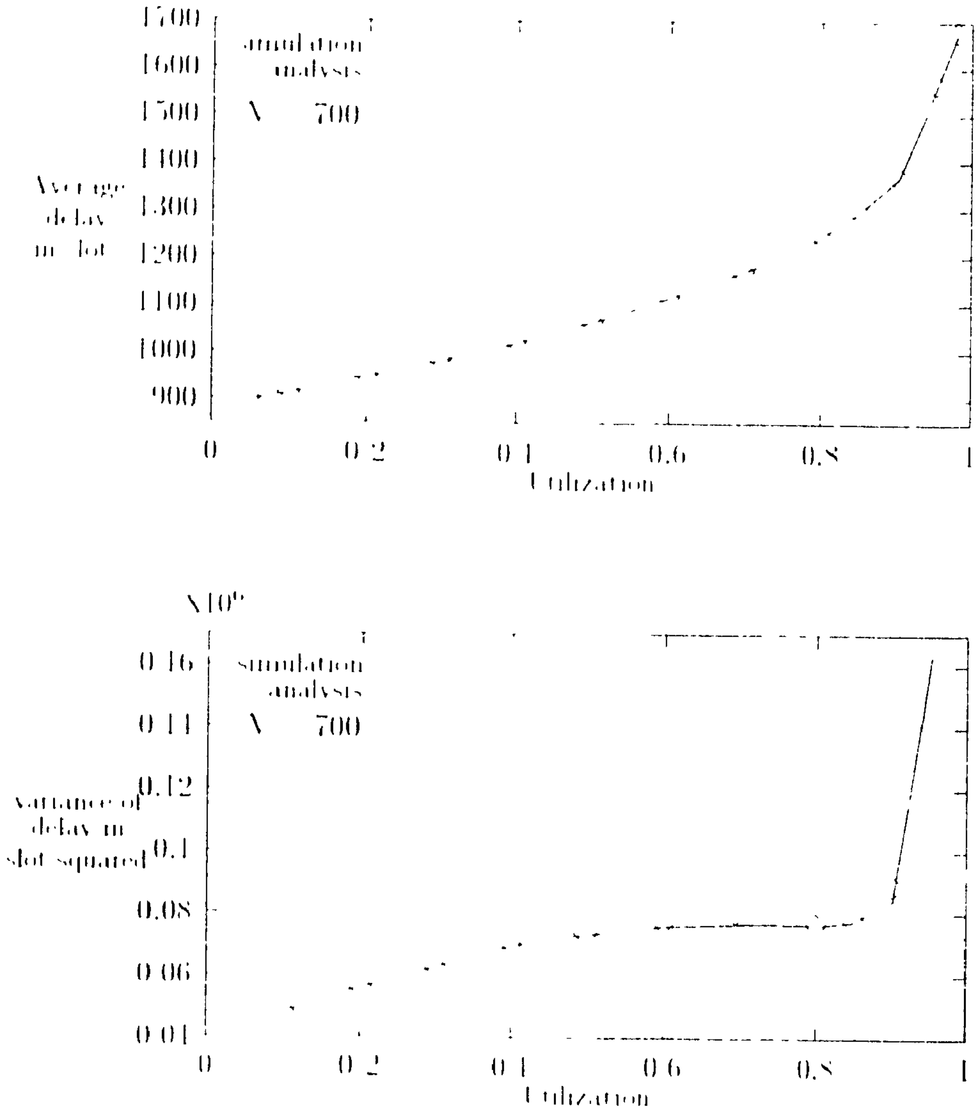


Figure 4.10 Comparison of analytic and simulation results

Chapter 5

Numerical Examples and Comparison

The following parameter values were chosen to characterize the wideband satellite channel and interactive user terminals. The channel bandwidth is 2.048 Mb/s and the user terminals generate single packet messages of length 1024 bits. Hence Data Slot length is $\frac{1024}{2.048X10^6} == 500\mu sec$. The Res Slot length is assumed to be of 6% of Data Slot, which is $30\mu sec$. The round-trip propagation delay is 0.27sec 540 Data Slots.

As we saw in Figure 3.6, PB version of CFDAMA performs better than other versions, hence, in this comparison CFDAMA PB is chosen to represent the class of CFDAMA protocols.

The CFDAMA was simulated for various values of λ. The performance is evaluated in terms of the average packet delay, the variance of packet delay and the cumulative probability distribution function of the packet delay. The performance is compared with FDMA, TDMA Reservation and CRRMA with multi request handler [23], [24]. In [24], the CRRMA performance was evaluated neglecting the channel overhead due to reservation. For fair comparison we simulated the CRRMA with reservation overhead taken into account. The length of reservation minislot is assumed to be 6% of data slot. The average delay response of the four protocols are compared in Figure 5.1. The CFDAMA has better average delay than TDMA and TDMA Reservation for all N over the entire utilization range. Compared to CRRMA, the CFDAMA has better delay at mid and high utilization ranges. The average delay curve of CRRMA is quite insensitive to N (and is hence suitable for infinite population). The CFDAMA average delay curve moves up with increasing N, nevertheless even for an user population as high as 700, it outperforms CRRMA right from an utilization of 0.5.

Variance of packet delay is an indication of how widely a packet's typical delay could deviate from the predicted average value. Figure 5.2 shows the performance of all the above four schemes in terms of variance of packet delay. For TDMA and CRRMA the variance explode as the channel utilization rises. While TDMA Reservation and CFDAMA have a smaller and comparable variance and is almost constant over the entire utilization range.

The cumulative probability distribution function (CDF) is a new measure introduced here. This gives the system designer the important information of system reliability, the reliability in meeting a certain delay bound in delivering a message to a destination. That is, how reliably (or with what probability) a messages could meet a delay constraint at a given channel utilization. Figure 5.3 and Figure 5.4 shows the CDF of various schemes at a low and high utilization point of 0.4 and 0.8 respectively. The CDF for TDMA and CFDAMA are smooth raising curves; CR-RMA has a step form, owing to the collision and subsequent retransmission nature of the scheme; TDMA Reservation has a ramp since packets arrival within a frame is uniformly distributed. From the plotted CDF, we have deduced the delay constraint (in number of slots) that can be met with 90% reliability and have tabulated the results in Table 5.1. In otherwords this table shows the delay within which 90% of a stations packets are bounded. The CFDAMA meets a much lower delay constraint than the others, especially when the channel utilization is high.

Finally, it is also noteworthy that CFDAMA-PB is very easy to implement. It has a single simpler slot structure unlike TDMA Reservation and CRRMA. TDMA Reservation and CRRMA have two types of slots, viz. reservation minislot and data slot and requires the user station to generate two types of packets (data packet and reservation packet) and be able to access these slots independently. Moreover, CRRMA poses the notorious problem of reliably detecting the reservation packet collision. We could say, the ease of implementation of CFDAMA is comparable to that of TDMA. Since CFDAMA-PB does not use separate slot to make reservation, there is no reservation channel overhead. Thus the maximum channel utilization is *one*.

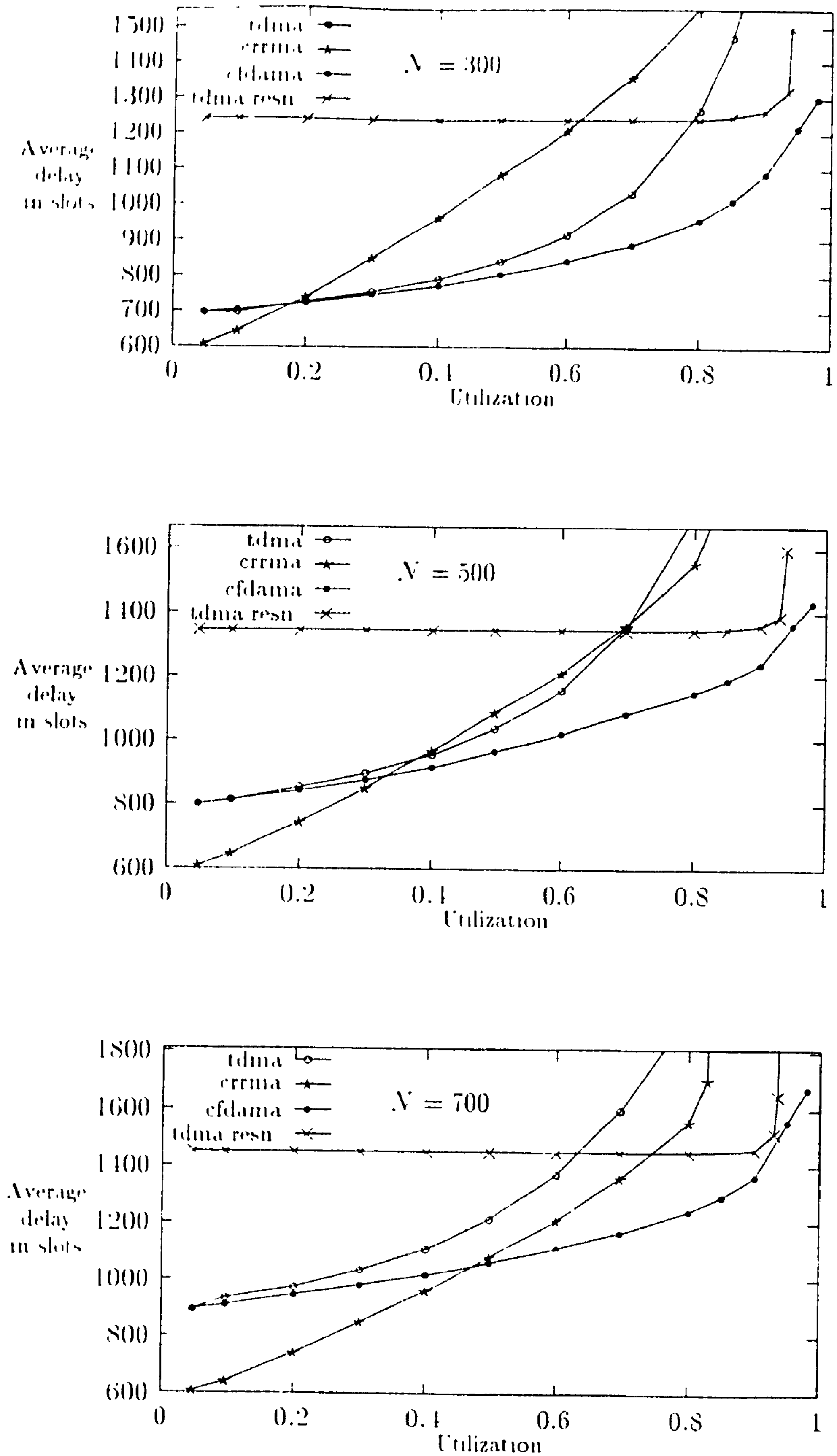


Figure 5.1: Average packet delay for various values of N.

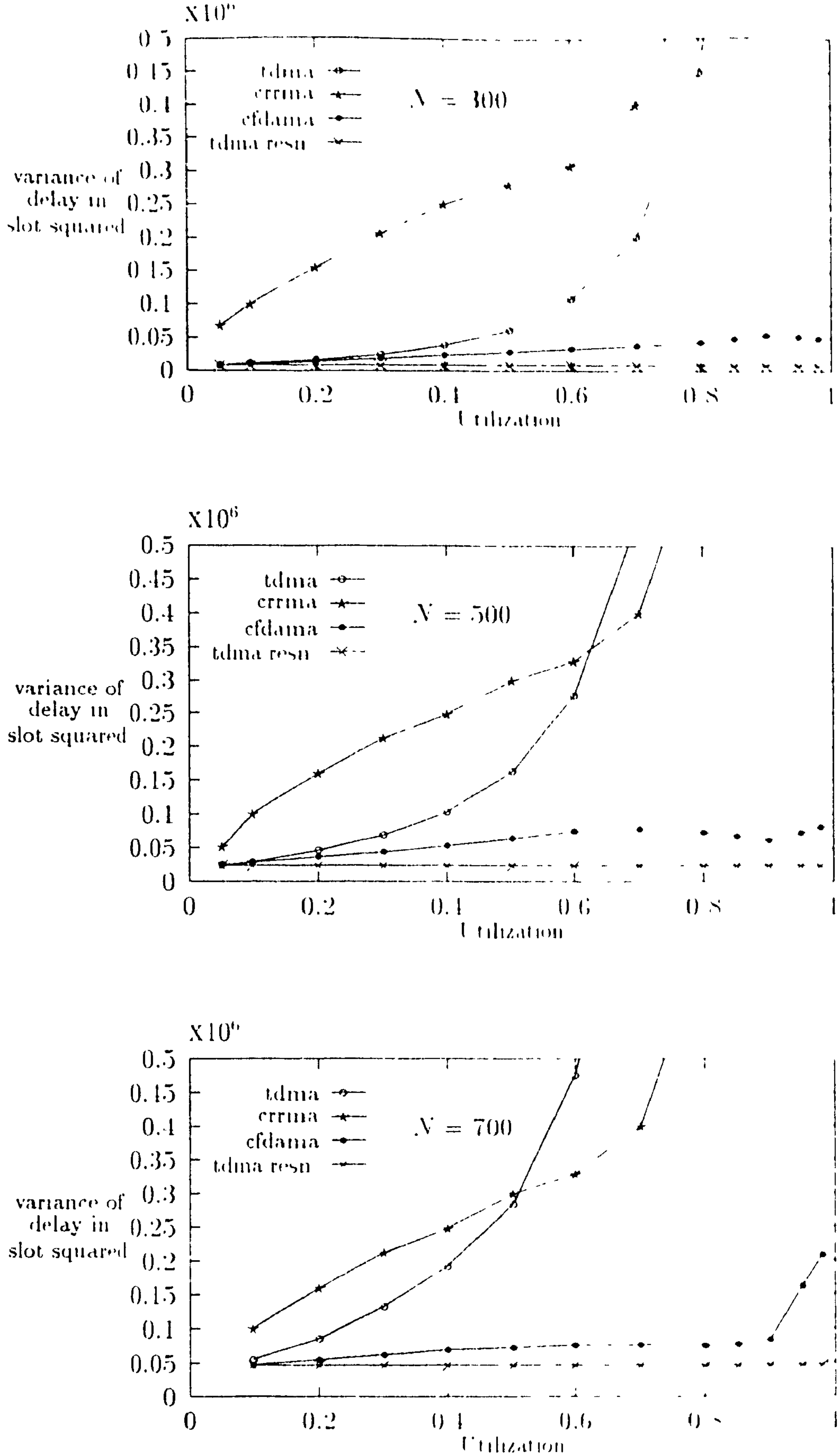


Figure 5.2: Variance of packet delay for various values of N

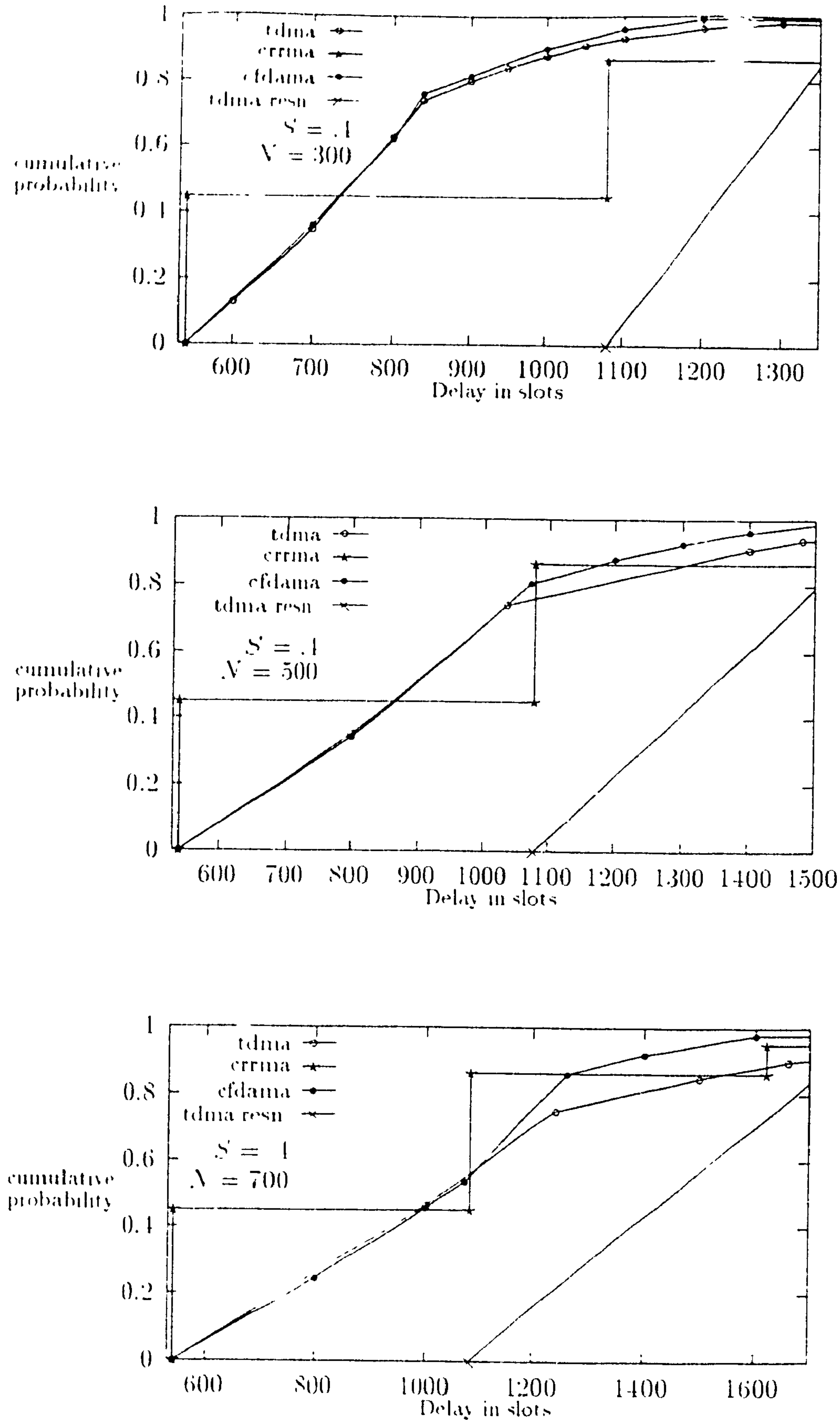


Figure 5.3: CDF of delay at 40% channel utilization for various N.

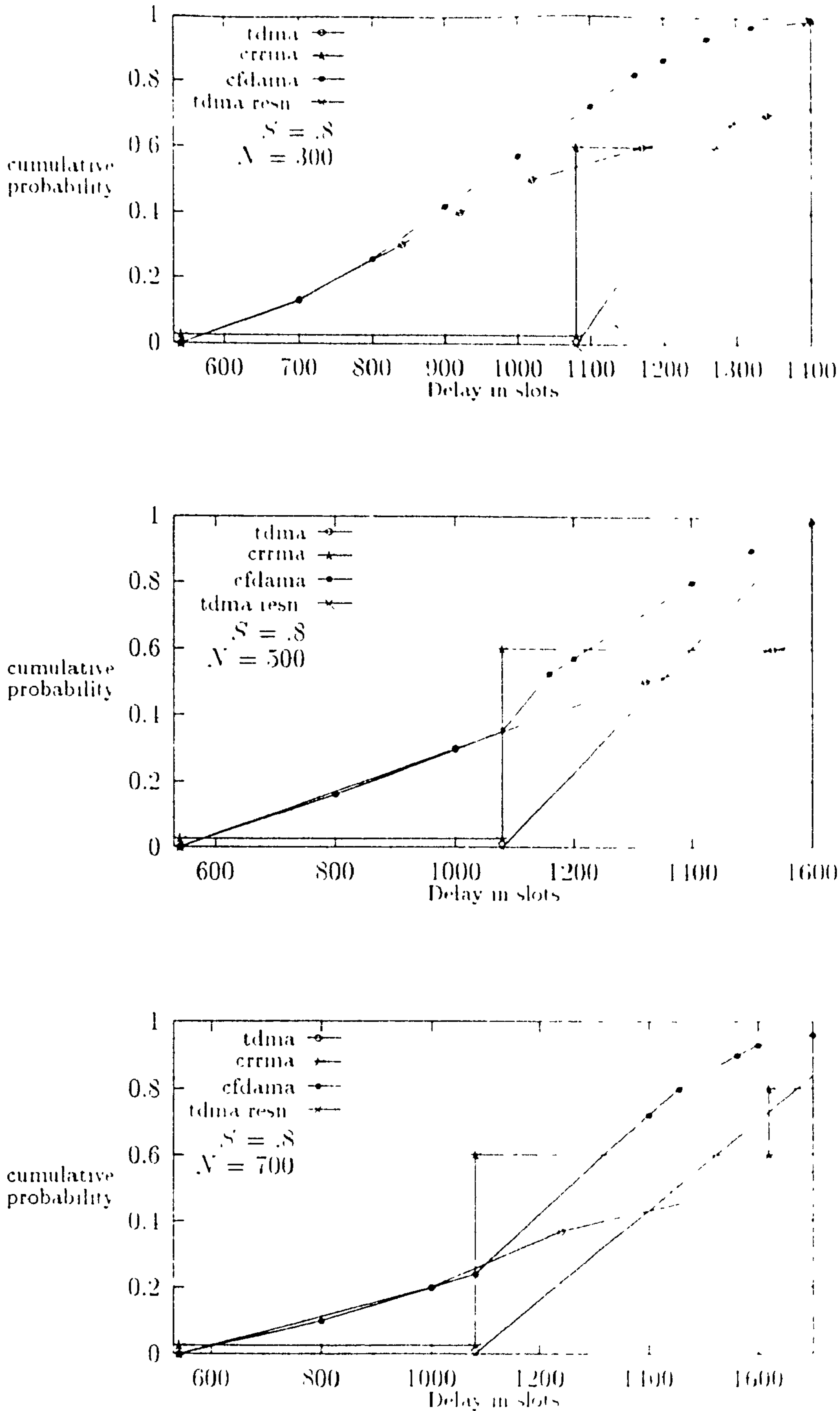


Figure 5.4: CDF of delay at 80% utilization for various N

scheme	$N = 300$		$N = 500$		$N = 700$	
	$S = .4$	$S = .8$	$S = .4$	$S = .8$	$S = .4$	$S = .8$
cfdama	1000	1232	1250	1500	1343	1562
tdma	1030	1603	1401	2859	1660	3544
tdma resvn.	1366	1369	1557	1560	1748	1752
crma	1620	2701	1620	2700	1619	2700

Table 5.1: Delay constraint that could be met with 90% reliability

Chapter 6

Conclusion and Further Research

This research has been concerned with multiple access techniques for packet switched satellite networks. An extensive survey of existing multiple access protocols for satellite networks was presented. The protocols were divided into six categories based on the prime idea of the scheme, and prominent examples under each category were discussed. A new class of multiple access protocols called CFDAMA was proposed. The proposed protocol is potentially efficient for the existing need of organizing the large number of geographically dispersed terminals into a network, to share processing capabilities. CFDAMA combined the salient features of TDMA and Demand assignment schemes and, always performs better than either of them throughout the entire utilization range. Three versions of CFDAMA, each employing a different method of channel reservation was investigated.

Two versions of the proposed protocols viz., one employing fixed assignment reservation channel and the other employing piggybacked reservation were modeled and analyzed within a queuing theoretic framework. Use of approximations exploiting the system structure has been shown to be helpful in simplifying analysis. The analytical results were then compared with those obtained from simulation in order to support the validity of the various simplifying approximation.

The underlying idea of CFDAMA to achieve low delay at all channel loadings wa to make a dynamic mix of division of traffic being sent via free assigned slots and via demand assigned slots.

The performance of the proposed schemes and a few other existing ones were evaluated in terms three performance measures: Average delay Vs Utilization, Variance of packet delay and Cumulative distribution function of packet delay. Of which, the last one is a new useful measure introduced here; it gives information on the reliability of the system in meeting a certain delay bound at a given utilization point.

Suggestions for further Research

- The CFDAMA schemes, as any other scheme based on reservation access, can be easily modified to incorporate the circuit switching capability for stream type traffic. That is, once a reservation is made, channel time could be assigned periodically so that guaranteed access is assured until the request is canceled. Performance evaluation and analysis under such conditions requires further investigation.

- The CFDAMA is not as efficient as random access schemes at low utilization ranges. Methods of incorporating random access component into CFDAMA would result in a scheme that performs best at all utilization for any user population size. Though this would rip off the simplicity inherent in CFDAMA, this strategy would become important if extreme efficiency is to be maintained.

- Performance may also improve when random access and piggybacking are allowed to coexist. This requires further study.

Bibliography

[1] Abramson, N., " The ALOHA System- Another Alternative for Computer Communications," *1970 Fall Joint Comput. Conf., AFIPS Cof. Proc.*, VOL. 37, AFIPS Press, Montvale, N.J., 1970

[2] Ahmadi, H. and Stern, T.E., "A New Satellite Multiple Access Technique for Packet Switching Using Combined Fixed and Demand Assignments," *Conf Records of NTC*, pp. 70.4.1-70.4.3, 1980.

[3] Balagangadhar, M.M and Pickholtz, R.L., " Analysis of a Reservation Multiple Access Techniques for Data Transmission Via Satellite" *IEEE Trans. on Commun.*, VOL. COM-27, pp 1467-1475, Oct 1979

[4] Binder, R., "A Dynamic Packet Switching System for Satellite Broadcast Channels ," *Proceedings of ICC*, pp. 41.1-41.5, 1975

[5] Borgonovo, F. and Fratta, L.,"SRUC: A Technique for Packet Transmission on Multiple Access channels," *Proceedings of the FOURTH ICCC*, pp. 601-608, 1980.

[6] J.I. Captanakis, "Tree Algorithms for Packet Broadcast channels," *IEEE Trans. Inform. Theory*, VOL. C-31, pp 505-515, Sept. 1979

[7] Cox D R, *Renewal Theory*, Methuen(London) 1962

[8] DeRosa, J.K., Ozarow, L.H. and Weiner, L.N., " Efficient Packet Satellite Communications," *IEEE Trans. on Commun.*, VOL. COM 27, pp 1116-1122, Oct 1979.

[9] Gagliardi R. M. *Satellite Communications*, Van Nostrand, Rainhold, New York, 1991

[10] G Falk *et al.*, " Integration of Voice and Data in the Wideband Packet Satellite Network," *IEEE J Select.Areas Commun.*, VOL. SAC-1,No.6, Dec.1983.

[11] Green P E. Jr , ed., "*Computer Network Architecture and Protocols*," New York, Plenum Press, 1982.

[12] J F Hayes, "An Adaptive Technique for Local Distribution," *IEEE Transactions on Commun* VOL. COM-26 pp 1178-1186, Aug. 1978

[13] J F Hayes, and M.Mehmet Ali, "Random Access Systems: ALOHA's Progeny", *Can.J Elect and Comp Eng.* VOL. 14 pp 3-10, 1989

[14] J F Hayes, "*Modeling and Analysis of Computer Communication Networks*", Plenum Press, 1984.

[15] H Bruneel, "Mesaage Delay in TDMA Channels with Contiguous Output ", *IEEE Transactions on Commun* VOL COM-34 pp 681-684, july 1986.

[16] Jacobs I M, Binder R. and Hoversten E.V., " General Purpose Packet Satellite Networks ," *Proc of the IEEE*, VOL. 66, No.11, Nov. 1978

[17] J.N Pelton and W.W Wu, "The Challenge of 21st Century Satellite Communications INTELSAT enters the Second Millenium.", *IEEE J.Select.Areas Commun.*, VOL SAC-5, No.4, May 1987

[18] L Kleinrock, "*Queuing Systems*" VOL. 1: Theory,John Wiley, 1975

[19] L. Kleinrock and S. S. Lam, "Packet Switching in a Multiaccess Broadcast Channel: Performance Evaluation," *IEEE Trans. on Commun.*, COM-23(4), 410-422, April 1975.

[20] L. Kleinrock, and Y. Yemini, "An Optimal Adaptive Scheme for Multiple Access Broadcast Communications," *Proc. of ICC*, pp 7.2.1-7.2.5, June 1978.

[21] L. Kleinrock, "On Resource Sharing in a Distributed Communication Environment," *IEEE Communications Magazine*, VOL 17, pp 27-31, Jan 1979.

[22] S. Lam and L. Kleinrock, "Packet Switching in a Multi access Broadcast Channel: Dynamic Control Procedures," *IEEE Trans. on Commun.*, COM-23, pp 891-905, April 1975.

[23] H.W. Lee and J.W. Mark, "Combined Random/Reservation Access for Packet Switched Transmission over a satellite with On Board Processing: Part I Global Beam Satellite," *IEEE Trans. on Commun.*, VOL. COM-31, pp 1161-1171, Oct 1983.

[24] H.W. Lee, "*Hybrid Random/Reservation Access Protocols for Communication Satellites*," Ph.D Thesis, University of Waterloo, 1984.

[25] T. Le-Ngoc and Y. Yao, "CREIR: A Multiple Access Protocol for On Board Processing Satellite Systems," *Can. Conf. on ECE* 1991, Quebec.

[26] T. Le-Ngoc and J.I. Mohammed, "Combined Free/Demand Assignment Protocols for Wideband Packet Satellite Communications," to appear in *ICUPC '93*, Ottawa.

[27] L.T. Liu and D. Towsley, "Window and Tree Protocols for Satellite Channels," *Proc. IEEE Infocom*, pp 215-221, 1983.

[28] J.L. Massey, "Collision Resolution Algorithms and Random Access Communications," in *Multiuser Communication Systems*, G Longo, Ed. New York: Springer Verlag, 1981, pp 73-137

[29] Nuspl, P.P, Brown, K.E, Steenaart, W. and Ghicopoulos, "Synchronization Methods for TDMA," *Proceedings of IEEE*, VOL. 65 pp 131-144, Mar. 1977

[30] Raychaudhuri D, "Announced Random Retransmission Multiple Access Protocols," *IEEE Trans. on Commun.*, july 1984

[31] Rubin I, "Message Delays in FDMA and TDMA Communication Channels," *IEEE Trans. Commun.*, VOL. COM-27, No. 5, May 1979.

[32] W G Schmidt, "Satellite Time-Division Multiple Access Systems: Past, Present and Future," *Telecommun.*, vol. 7, pp. 21-24, Aug. 1974.

[33] Shuji Tasaka, "*Performance Analysis of Multiple Access Protocols*" The M.I.T press, 1986.

[34] F.A Tobagi, "Packet Switching in Radio Channels: Part III - Polling and Split channel Reservation Multiple Access Protocol," *IEEE Trans. on Commun.*, Aug. 1976

[35] F.A Tobagi, M Gerla, R.W Peebles, and E.G Manning., "Modeling and Measurement Techniques in Packet Communication Networks," *Proc. IEEE*, VOL. 66, pp 1423-1447, Nov. 1978

[36] Tsybakov, B.S, "Survey of USSR Contributions to Random Multi-Access Communications," *IEEE Trans. Inform. Theory*, VOL. IT-31, No. 2, pp. 143-166, March 1985.

[37] Viterbi A.J, "A Perspective on the Evaluation of Multiple Access Satellite Communications," *IEEE J Select. Areas Commun.*, VOL 10, No 6, August 1992

[38] Wieselthier, J.E and Ephremides, A. "A Class of Protocols for Multiple Access in Satellite Networks." *IEEE Trans. Automat. Contr.* VOL. AC-25, No. 5, pp 865-879, Oct. 1980.

[39] E.W.M Wong and T. Yum "A Controlled Multiple Access Protocol for Packet Satellite Communication." *IEEE Trans. on Commun.*, VOL. COM-39, pp 1133-1140, july 1991

Appendix A

Evaluation of $\check{\lambda}$

The User ID has to advance N steps up the table in order to release a free user. On the average, for the first $R(1-\check{\lambda})$ number of steps the first and the second arrival will not hinder the advancement if the forthcoming server is a free server and the first arrival alone will not hinder if the forth coming server is a called server. Hence, for the first $R(1-\check{\lambda})$ steps, we have,

$$\Pr(\text{the tagged pkt unhinder}|E_f) = \sum_{j=N}^{\infty} \frac{j\check{l}_j}{\check{L}'(1)} \sum_{i=1}^{j} \frac{1}{j}(e^{-\lambda(j-i)} + \lambda(j-i)e^{-\lambda(j-i)})$$

$$= \frac{e^{\lambda}}{\check{L}'(1)}\left(\frac{(1-\lambda)(\check{L}(e^{-\lambda})-1)}{1-e^{\lambda}} - \lambda\frac{(\frac{d}{d\lambda}\check{L}(e^{-\lambda}) + \check{L}'(1))}{1-e^{\lambda}} + \lambda\check{L}'(1)\right)$$

$$\begin{aligned}\Pr(\text{the tagged pkt unhinder}|E_c) &= \sum_{j=R}^{\infty} \frac{jl_j}{L'(1)} \sum_{i=1}^{j} \frac{1}{j} e^{-\lambda(j-i)} \\ &= \frac{e^{\lambda}L(e^{-\lambda})-1)}{L'(1)(1-e^{\lambda})}\end{aligned}$$

$$\left.\begin{array}{l}\text{Avge no. of arrvls per slot (during first} \\ R(1-\check{\lambda})\text{ steps) that will unhinder, } \lambda_{uh}\end{array}\right\} = N\lambda[\Pr(E_c).\Pr(\text{the tagged pkt unhinder}|E_c)$$

$$+\Pr(E_f).\Pr(\text{the tagged pk unhinder}|E_f)]$$

$$\left.\begin{array}{l}\text{Avge no. of arrvls per slot (during first} \\ R(1-\check{\lambda})\text{ steps) that will hinder, } \lambda_h\end{array}\right\} = N\lambda - \lambda_{uh}$$

For the rest of the $N - R(1 - \tilde{\lambda})$ number of steps, the average arrival rate of requests that will hinder the advancement is exactly same as λ_s. Therefore the average arrival rate of requests that will effectively hinder the advancement of the User ID is given by

$$\tilde{\lambda} = \lambda_h \frac{R(1 - \tilde{\lambda})}{N} + \lambda_s \frac{N - R(1 - \tilde{\lambda})}{N}$$

$\tilde{\lambda}$ can be obtained as a numerical solution to the above equation. The obtained solution is found to comply with the simulation results.

Appendix B

Simulation Models

Simulation was done using a software package called OPNET (Optimized Network Engineering Tools), a powerful tool used to simulate large communication networks with detailed protocol modeling and analysis. OPNET provides a window with a very easy to use graphical interface. With the help of graphical interface we can specify the network model, node model and the process models in a top to down straightforward fashion.

In the following is given the modeling details of CFDAMA-PB. The modeling methodologies employed for other versions of CFDAMA are quite similar to the described PB version and are hence not included.

The network model is shown in Figure B.1; The node at the center of the figure is the satellite node and the surrounding nodes (node n0 to node n9) represents the user groups. The number of users in each group is a variable parameter to be specified at the simulation time. By promoting this variable to be specified at the simulation time we can repeat the simulation over a range of user population size and collect the statistics that vary with user population size. Since we have defined the number of groups to be ten we can increment the number of users only in integral multiples of ten.

Each user node consists of a packet generator. The packet generator module, an available module of OPNET library, is given with the following parameters (i) inter packet generation time (exponentially distributed) and (ii) packet length (1024 bits). This module generates packets according to the specified parameters and transfers the packet to one of the user queue randomly this random assignment of packets mimics as though each user independently generate its own proportion of packets with exponentially distributed inter generation time. The user queue module is a passive FIFO queue. Queues that do not contain a server are called passive queues (active queues come with a built in server with user defined service rate).

In each group there are a number of users. Each user is provided with a separate queue. Each user (and hence each queue) is identified using an ID which has two parts: the group number and the individual user ID within the group. The satellite node has a clock which triggers a process at intervals of Data Slot length. During every process the node checks its queue (the OBS queue), if there is any request, it picks up the request at the head of the queue, examines the group ID and the individual user ID to find out from whom the request have originated. Once it finds this information it issues a number of data slots equivalent to the requested number to the user. The issued slot arrives at the earth (at all user nodes) after an elapse of 135 milli-second.

If the scheduler queue is empty the OBS issues the slot to the user ID at the top of the stack; subsequently the user ID is moved to the bottom of the stack (the round robin scheme of free slot assignment or User ID reordering is achieved in this fashion).

Users generate packets according to a Poisson process with a mean arrival rate specified during the simulation time. whenever a packet is generated the current simulation time is entered into the packet and the packet is deposited in its queue. When the user receive slot-interrupt from the satellite, it takes the packet at the head of the queue and, using the current time and the time when packet was generated it computes the delay the packet has experienced. Consequently it enters the computed delay into the global statistics collection file and finally destroys the packet.

At the end of the simulation the satellite node computes the required performance measures and writes out in the form of graphs.

The details of modeling extracted from the OPNET is attached. Figure B.2 shows the satellite node model and the associated processes. The processes are given in the form of a Finite State Machine diagram. In Figure B.3 is shown the user group nodes and the associated processes.

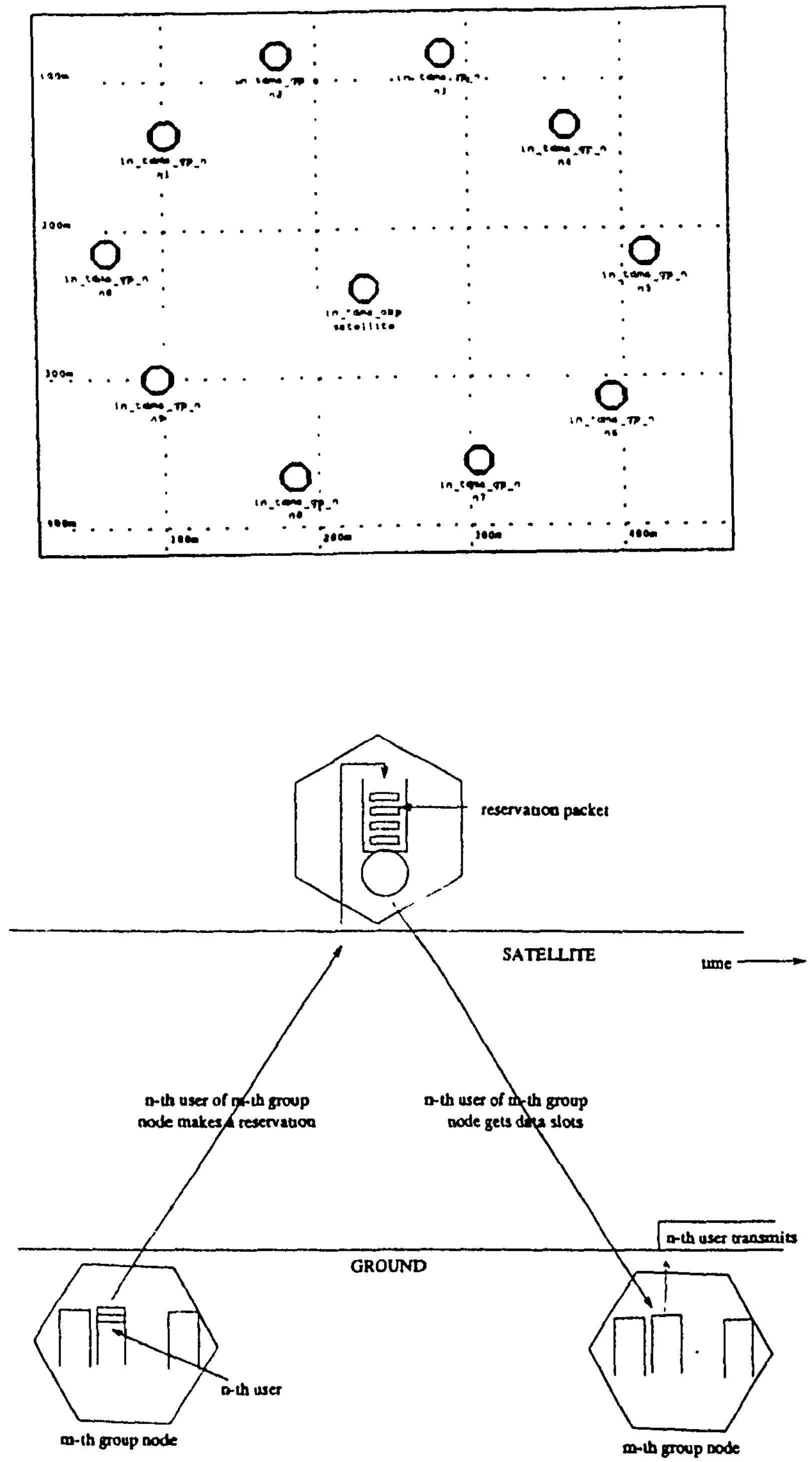


Figure B.1: Network Model

top subnet

subnet
tdma_res

Attributes	
Attr Name	*Attr Value*
name	tdma_res
user id	1
sys id	1
latitude	0 6 (deg)
longitude	89 1 (deg)
icon	subnet

fixed node
n0

Attributes	
Attr Name	*Attr Value*
name	n0
model	in_tdma_gp_n
user id	0
sys id	0
condition	enabled
x position	63 2812 (m)
y position	214 844 (m)
altitude	0 (m)
icon	fixed comm

fixed node
n1

Attributes	
Attr Name	*Attr Value*
name	n1
model	in_tdma_gp_n
user id	1
sys id	1
condition	enabled
x position	101 562 (m)
y position	135 938 (m)
altitude	0 (m)
icon	fixed comm

fixed node
n2

Attributes	
Attr Name	*Attr Value*
name	n2
model	in_tdma_gp_n
user id	2
sys id	2
condition	enabled
x position	175 (m)
y position	82 8125 (m)
altitude	0 (m)
icon	fixed comm

fixed node
n3

Attributes	
Attr Name	*Attr Value*
name	n3
model	in_tdma_gp_n
user id	3
sys id	3
condition	enabled
x position	282 031 (m)
y position	82 0312 (m)
altitude	0 (m)
icon	fixed comm

fixed node
n4

Attributes	
Attr Name	*Attr Value*
name	n4
model	in_tdma_gp_n
user id	4
sys id	4
condition	enabled
x position	364 062 (m)
y position	130 469 (m)
altitude	0 (m)
icon	fixed comm

fixed node
n5

Attributes	
Attr Name	*Attr Value*
name	n5
model	in_tdma_gp_n
user id	5
sys id	5
condition	enabled
x position	414 844 (m)
y position	215 625 (m)
altitude	0 (m)
icon	fixed comm

fixed node
n6

Attributes	
Attr Name	*Attr Value*
name	n6
model	in_tdma_gp_n
user id	6
sys id	6
condition	enabled
x position	392.188 (m)
y position	313.281 (m)
altitude	0 (m)
icon	fixed_comm

fixed node
n9

Attributes	
Attr Name	*Attr Value*
name	n9
model	in_tdma_gp_n
user id	9
sys id	7
condition	enabled
x position	95.3125 (m)
y position	300 (m)
altitude	0 (m)
icon	fixed_comm

fixed node
n8

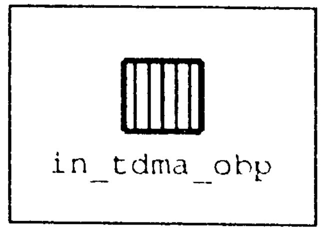
Attributes	
Attr Name	*Attr Value*
name	n8
model	in_tdma_gp_n
user id	8
sys id	8
condition	enabled
x position	184.375 (m)
y position	366.406 (m)
altitude	0 (m)
icon	fixed_comm

fixed node
n7

Attributes	
Attr Name	*Attr Value*
name	n7
model	in_tdma_gp_n
user id	7
sys id	9
condition	enabled
x position	305.469 (m)
y position	356.25 (m)
altitude	0 (m)
icon	fixed_comm

fixed node
satellite

Attributes	
Attr Name	*Attr Value*
name	satellite
model	rn_tdma_obp
user id	10
sys id	10
condition	enabled
x position	230 469 (m)
y position	239 844 (m)
altitude	0 (m)
icon	fixed_comm



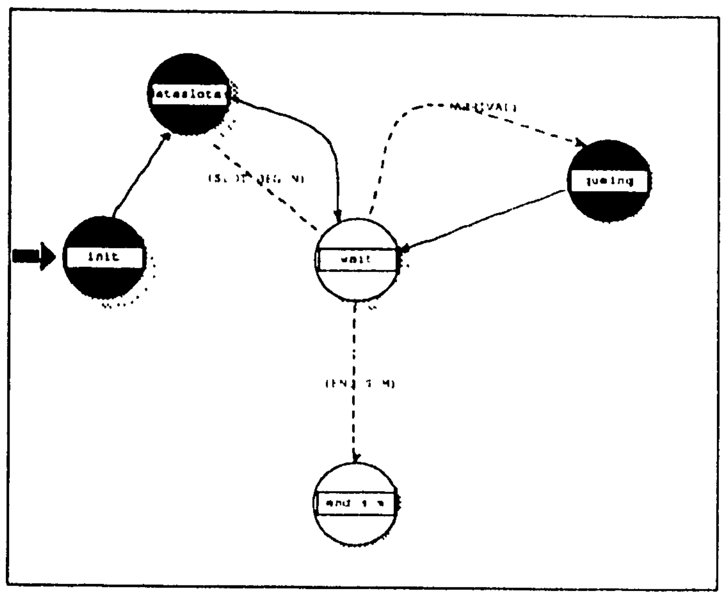


Figure B.2: Satellite node model and associated processes

Summary				
Number of States	*Header Block*	*State Variables*	*Temporary Variables*	*Function Block*
5	**Yes**	**Yes**	**Yes**	**No**

Header Block *13 lines*

```
#define SLOT_BEGIN op_intrpt_type() == OPC_INTRPT_SELF
#define END_SIM op_intrpt_type() == OPC_INTRPT_ENDSIM
#define ARRIVAL op_intrpt_type() == OPC_INTRPT_STRM
extern int USER_GROUP_SIZE
#define NO_OF_GROUPS 10
#define OBP_IN_STRM 0
 *global variables*
int no_pk_success, total_gen_pkts
double acc_pk_delay, sq_acc_pk_delay
extern double packet_size_tot
Objid obp_id,obp_node_id,subnet_id
extern double packet_size
```

State Variables *2 lines*

```
int \gp_id\count \user_id \free_slot_owner \free_slot_owner_gp
double \chnl_cap, \s_tr,
```

Temporary Variables *4 lines*

```
Packet *pk
Objid origin_obj_id, node_id, node_q_id, free_slot_owner_gp_id, free_slot_owner_gp_q_id
int slot_owner
double packet_delay, create_time,
```

State 0, init (Enter Execs) *forced, 14 lines*

```
gp_id = 0,
count = USER_GROUP_SIZE-1
obp_id = op_id_self()
obp_node_id = op_id_parent( op_id_self()),
subnet_id = op_id_parent(obp_node_id),
no_pk_success = 0,
free_slot_owner = 0,
free_slot_owner_gp = 0,
sq_acc_pk_delay =0
acc_pk_delay =0,
```

State 0, init (CET's)

CET #0	*Cond*	(1)
	Exec	
	Trans	**dataslotas**

State 1:	**wait (Enter Execs)**	*unforced, 0 lines*

State 1.	**wait (CET's)**	
CET #0	*Cond*	(SLOT_BEGIN
	Exec	.
	Trans	**dataslotas**
CET #1	*Cond*	(END_SIM
	Exec	.
	Trans	**end_sim**
CET #2	*Cond*	(ARRIVAL
	Exec	.
	Trans	**queing**

State 2:	**dataslotas (Enter Execs)**	*forced, 28 lines*

```
op_intrpt_schedule_self(op_sim_time()+packet_size,0);
if (!op_subq_empty (0))
{
pk = op_subq_pk_remove(0, OPC_QPOS_HEAD);
origin_obj_id = op_pk_stamp_mod_get(pk);
op_pk_nfd_get(pk,"...", &slot_owner);
op_intrpt_schedule_remote(op_sim_time()+0.135,slot_owner,origin_obj_id);
op_pk_destroy(pk);
}
else

free_slot_owner_gp_id = op_id_from_userid(subnet_id,OPC_OBJTYPE_NODE_FIXED,free_slot_owner_gp);
free_slot_owner_gp_q_id = op_id_child (free_slot_owner_gp_id,OPC_OBJTYPE_QUEUE,0);
op_intrpt_schedule_remote(op_sim_time()+0.135,free_slot_owner,free_slot_owner_gp_q_id);

if (free_slot_owner == USER_GROUP_SIZE-1)
   {
   free_slot_owner = 0;
   ++free_slot_owner_gp;
   }
else
   ++free_slot_owner;

     if(free_slot_owner_gp == NO_OF_GROUPS)
       free_slot_owner_gp = 0;
     else
       ;
}
```

State 2.	**dataslotas (CET's)**	
CET #0	*Cond*	(1)
	Exec	.

Trans **wait**

State 3 end_sim (Enter Execs) *unforced, 5 lines*

```
op_stat_write_scalar("[illegible]" (packet_size/tat));
op_stat_write_scalar("[illegible] Delay [illegible]" (acc_pk_delay/no_pk_success));
var = sq_acc_pk_delay/no_pk_success-(acc_pk_delay/no_pk_success)*(acc_pk_delay/no_pk_success)
op_stat_write_scalar("[illegible]" var)
}
```

State 3 end_sim (CET's)

State 4 queing (Enter Execs) *forced, 3 lines*

```
pk = op_pk_get(OBP_IN_STRM)
op_subq_pk_insert(0, pk, OPC_QPOS_TAIL)
```

State 4 queing (CET's)

CET #0	*Cond*	(1)
	Exec	
	Trans	**wait**

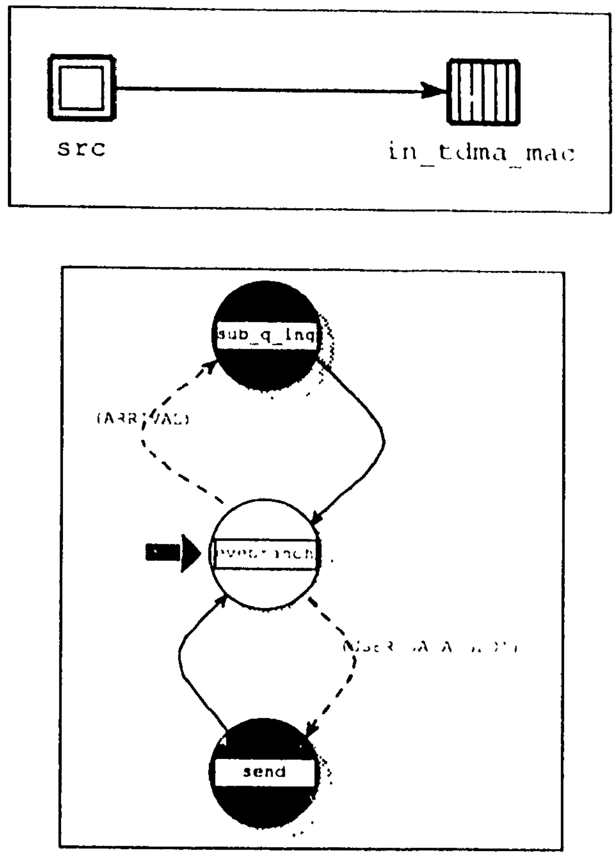


Figure B.3: Group node model and associated processes

Summary				
Number of States	*Header Block*	*State Variables*	*Temporary Variables*	*Function Block*
3	Yes	No	Yes	No

Header Block — *11 lines*

```
#define ARRIVAL                               \
    (op_intrpt_type() == OPC_INTRPT_STRM)
#define USER_DATA_SLOT                         \
    (op_intrpt_type() == OPC_INTRPT_REMOTE)
extern int USER_GROUP_SIZE
extern Objid obj_id, obj_node_id, subnet_id
#define ORP_IN_STRM_INDEX 0
extern double acc_pk_delay_sq, acc_pk_delay
extern int rc_pk_success
extern double packet_size
```

Temporary Variables — *4 lines*

```
Packet *pkptr, *pk, *respkptr;
int user_index, slot_owner;
double packet_delay, create_time;
int num_in_subq, i;
```

State 0: evebranch (Enter Execs) — *unforced, 0 lines*

State 0: evebranch (CET's)

CET #0	*Cond*	(ARRIVAL)
	Exec	.
	Trans	**sub_q_ing**
CET #1	*Cond*	(USER_DATA_SLOT)
	Exec	
	Trans	**send**

State 1: sub_q_ing (Enter Execs) — *forced, 5 lines*

```
pkptr = op_pk_get(op_intrpt_strm());
op_pk_stamp(pkptr);
op_pk_nfd_get(pkptr, "user_index", &user_index);
op_subq_pk_insert(user_index, pkptr, OPC_QPOS_TAIL);
```

State 1: sub_q_ing (CET's)

CET #0	*Cond*	(1)
	Exec	

Trans **evebranch**

State 2: send (Enter Execs) *forced, 42 lines*

```
slot_owner = op_intrpt_code();
if(!op_subq_empty(slot_owner+USER_GROUP_SIZE))
  {
  pk = op_subq_pk_remove (slot_owner+USER_GROUP_SIZE, OPC_QPOS_HEAD);
  op_pk_nfd_get(pk, "create_time", &create_time);
  packet_delay = op_sim_time() + 0.27 + packet_size - create_time;
  sq_acc_pk_delay += packet_delay*packet_delay;
  /* op_stat_write_global(0, "packet delay",packet_delay); */
  acc_pk_delay += packet_delay;
  ++no_pk_success;
  op_pk_destroy (pk);
  }
else
  {
  if(!op_subq_empty(slot_owner))
    {
    pk = op_subq_pk_remove (slot_owner, OPC_QPOS_HEAD);
    op_pk_nfd_get(pk, "create_time", &create_time);
    packet_delay = op_sim_time() + 0.27 + packet_size - create_time;
   /* op_stat_write_global(0, "packet delay", packet_delay); */
    sq_acc_pk_delay += packet_delay*packet_delay;
    acc_pk_delay += packet_delay;
    ++no_pk_success;
    op_pk_destroy (pk);
    }
  else
    ;
  }
if (!op_subq_empty (slot_owner))
  {
   num_in_subq = op_subq_stat (slot_owner, OPC_QSTAT_PKSIZE);
   for(i=0;i<num_in_subq;++i)
     {
     pkptr = op_subq_pk_remove(slot_owner,OPC_QPOS_HEAD);
     respkptr = op_pk_copy (pkptr);
     op_subq_pk_insert (USER_GROUP_SIZE + slot_owner,respkptr,OPC_QPOS_TAIL);
     op_pk_deliver_delayed (pkptr,obp_id,0,0.135);
     }
  }
else
  ;
```

State 2: send (CET's)

CET #0	*Cond*	(1)
	Exec	;
	Trans	**evebranch**