UNIVERSITY OF ALBERTA

Collision-Avoidance Star Local Area Networks:

In-Depth Performance Analysis with Applications to Priority and Real Time

By

Hossam S. Hassanein

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE DEGREE

OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTING SCIENCE

EDMONTON, ALBERTA

(Fall) 1990

National Library of Canada
Bibliothèque nationale du Canada

Canadian Theses Service
Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

ISBN 0-315-65035-4

UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR: HOSSAM S. HASSANEIN

TITLE OF THESIS: Collision-Avoidance Star Local Area Networks: In-Depth Performance Analysis with Applications to priority and Real Time

DEGREE: DOCTOR OF PHILOSOPHY

YEAR THIS DEGREE GRANTED: 1990

PERMISSION IS HEREBY GRANTED TO THE UNIVERSITY OF ALBERTA LIBRARY TO REPRODUCE SINGLES COPIES OF THIS THESIS AND TO LEND OR SELL SUCH COPIES FOR PRIVATE, SCHOLARLY OR SCIENTIFIC RESEARCH PURPOSES ONLY.

THE AUTHOR RESERVES OTHER PUBLICATION RIGHTS, AND NEITHER THE THESIS NOR EXTENSIVE EXTRACTS FROM IT MAY BE PRINTED OR OTHERWISE REPRODUCED WITHOUT THE AUTHOR's WRITTEN PERMISSION.

(Student's Signature)

(Student's Permanent Address)

Date: Oct. 12, 1990

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

THE UNDERSIGNED CERTIFY THEY HAVE READ, AND RECOMMENDED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH FOR ACCEPTANCE, A THESIS ENTITLED "Collision-Avoidance Star Local Area Networks: An In-Depth Study with Applications to Priority and Real Time"

SUBMITTED BY Hossam S. Hassanein

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

W. Dobosiewicz

A. Kamal (Supervisor)

P. Gburzynski

M. T. Ozsu

J. Culberson

B. Schmuland

D. Eager

To my wife, *Salwa*

ABSTRACT

This thesis is concerned with the study of Collision-Avoidance Star Local Area Networks (CASLANs). CASLANs are a class of random access star local area networks in which packet collisions are avoided by using a number of hardware switches. The study is divided into three parts. The first is an indepth analysis of the behavior and performance of CASLANs. The second part involves analysis of CASLANs in a time-constrained environment. In the third part, prioritized CASLANs for data, and integrated voice/data applications are studied.

During the course of this study several contributions to the understanding of the behavior and performance of CASLANs are made:

It is shown that decreasing the propagation delay between the nodes and the central node does not always mean better performance. Indeed, it is shown that increasing the propagation delay may result in a much improved performance. Also, it is proven that by choosing the right network parameters, packets could be guaranteed transmission rights.

It is shown that CASLANs, under very heavy load, operate in a round robin fashion, and that the delay is bounded.

A study of CASLANs for real-time applications is conducted, in which it is shown that CASLANs perform very well in a time-constrained environment. As well, the performance of CASLANs in a prioritized environment is explored. It is shown that CASLANs are very good candidates for voice and data integration.

Also, several other contributions to modeling the performance of CASLANs are made:

An exact, heavy load, performance model of a class of CASLANs is introduced. This model is used to show that the delay of CASLANs, under heavy load conditions, is bounded.

A new performance model for CASLANs is introduced. The model follows a tagged user in an exact analysis, while the rest of the users' behavior is approximated by mimicking that of the tagged user at steady state. This model is more accurate than all earlier CASLANs performance models.

A hard real-time performance model of CASLANs is presented. In this model, packet laxities are assumed to be exponentially distributed and packets that exceed their laxities are removed from the system.

A prioritized CASLANs model for moderately loaded systems is introduced.

ACKNOWLEDGMENT

I would like to thank my supervisor Dr. Ahmed Kamal. His teaching and guidance will long live in my mind and his friendship will always be in my heart.

I would also like to thank the members of my examining committee, Dr. J. Culberson, Dr. D. Eager, Dr. P. Gburzynski, Dr. M. T. Ozsu and Dr. B. Schmuland. Their comments and suggestions helped improve the quality of this thesis.

The great moral support of my family, Saad, Hanem, Khaled and Naglaa, is very much appreciated. My father, Saad, has always set a role model to me as a university professor and as a family man. My mother, Hanem, deserves full credit for instilling the confidence and tenacity I needed to pursue my Ph.D. For my brother, Khaled, I say thank you for not only being such a fine brother, but also for being my best friend.

The joyful moments I spent with my son, Tariq, and late-comer daughter, Randa, have always been some of the most pleasant times I spent away from school.

I am also indebted to my friends Amr El-Sheikh and Khaled Obaia for being great "partners" and "good helpers", for always "calling" on me and for never "passing" when I "call" on them.

The help and support of the staff and graduate students of the department of computing science at the university of Alberta have been a great asset. Specially, I would like to thank Mr. Bandula Abeysundara, Mr. Robert Baron, Mr. Ashraf El-Nagar, and Ms. Edith Drummond.

The financial support that has been provided by the department of Computing Science at the University of Alberta, as well as the Natural Sciences and Engineering Research Council of Canada is greatly appreciated.

Last, but not least, I would like to thank my wife, Salwa, for her encouragement and understanding throughout my studies at the University of Alberta., and for putting up with me all the weekends in the past couple of years, going shopping at West Edmonton Mall instead.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

List of Notations

N	Number of nodes or users in the system.
T	Packet length.
τ	Retry time.
K	$\left\lfloor \frac{T}{\tau} \right\rfloor$
Δ	$T \bmod \tau$
λ	Packet arrival rate per bit time.
$D(\tau)$	Average delay with retry time τ.
offered load	Number of packets during an average packet transmission time and is given by $(N \times T \times \lambda)$.
t	The overhead period between two consecutive hub acquisitions.
$\bar{t}$	Average length of the overhead period (t).
Z	The time measured from the start of the hub acquisition to the arrival of the tagged user's packet.
X	The remainder of the hub acquisition time measured from the instant of the arrival of the tagged user's packet to the hub.

Y	The remainder of the hub acquisition time measured from the instant of the last intrapacket arrival of the tagged user's packet to the hub.
Z_2	The second interpacket arrival time.
D_Z	The remainder of a packet delay starting from the second interpacket arrival time.
μ	Packet loss rate per bit time.
g	Grid size
ζ	Slot size
σ	Probability that an idle user generates a packet during a slot.
α	Probability that a packet from a carry-over ready user is lost during a slot.
Idle user	A user with an empty buffer.
Ready user	A user with a packet to transmit.
First-time ready user	A user which becomes ready during the current hub acquisition. Also known as a *new* user.
Carry-over ready user	A user which was ready before the start of the current hub acquisition.

$\bar{R}$	The tagged user is in the idle state (not ready).
R	The tagged user is in the ready state.
A	The tagged user is acquiring the hub.
$\mathbf{P}$	Transition probability matrix.
$\vec{\pi}$	Steady state probability vector.
$\bar{O}$	Average overhead period.
U	System Utilization.
U_{tag}	Hub utilization by the tagged user.
$\bar{D}$	Average delay
Laxity	A packet laxity is the latest time to transmit the packet before it is considered lost.
Pr_{loss}	Probability that the tagged user's packet is lost.
P(A)	The proportion of packets transmitted by the tagged user.
$P_{tag}(t \mid R)$	Probability that the tagged user's packet arrives at the hub in slot t given that it was ready.
C_i	A class i in a prioritized CASLAN.
Hub hogging	The phenomenon where one class monopolies the use of the hub.

X_i	Throughput per time unit for class i.
D_i	Average delay of class i.
talkspurt	A period when a voice source is active.
Silenceperiod	A period when a voice source is inactive.
L_p	The length of the period between voice packet generations.
S_v	Voice coding rate (bit/sec).
T_v	Voice packet length.

Chapter 1

Introduction

A *computer network* is an interconnection of several entities that allows these entities to successfully exchange information amongst them. The entities can be any combination of host computers, terminals, communication equipment, or even another subnetwork. Because of the diversity of such entities, many different terms are commonly used to refer to them among which are *hosts*, *nodes*, *sites*, *sources*, and *users*.

Computer networks are often classified according to their geographic distribution. Two major classes of communication networks exist: Wide Area Networks (WANs) and Local Area Networks (LANs). WANs, which are sometimes referred to as *long haul networks*, are those networks that span more than a few kilometers, usually in the order of tens, hundreds, and sometimes thousands of kilometers and could span continents. On the other hand, LANs usually span a distance of just a few kilometers. Other than spanned distance, LANs differ from WANs in the following,

(1) Ownership by a single organization, e.g., a university or a company.

(2) Higher transmission rates, on the order of several to hundreds of Mbps (Mega bits per second).

(3) Routing techniques are usually not employed, since there is a unique, known a priori, path between any pair of nodes.

Another major advantage of LANs over WANs is the ability of each node to sense the state of a common broadcast channel before attempting to use it. This gave rise to a class of local area network protocols called ***multiple access protocols***.

1.1. Multiple Access Protocols

Multiple access communication networks are those networks in which users (nodes) share a common communication channel. Multiple access protocols fall into two major classes: random access protocols and controlled access protocols [1-4]. (See Figure 1.1). In random access protocols, all users (or at least a subset of users) may be given transmission rights at the same time. This gives rise to packet collisions, and consequently packet retransmissions. Controlled access protocols avoid collisions by coordinating channel access such that only one station attempts to transmit at a time.

1.1.1. Controlled Access Protocols

In controlled access protocols channel assignment to stations is either fixed or made upon demand. In fixed assignment, the channel is allocated to stations in a static manner. Time Division Multiplexing (TDM), Frequency Division Multiplexing (FDM) and Space Division Multiplexing are well known fixed assignment channel access schemes.

In demand assignment access schemes, stations do not assume transmission rights unless they have a packet to transmit. The access order could be implemented via token passing or by some predetermined order. Token passing could be explicit or implicit. In explicit token passing schemes such as the token ring [5] and the token bus [6], a station is allowed to transmit its packet only after receiving an explicit token. The token is released upon transmission completion. Implicit token passing schemes include BRAM (Broadcast

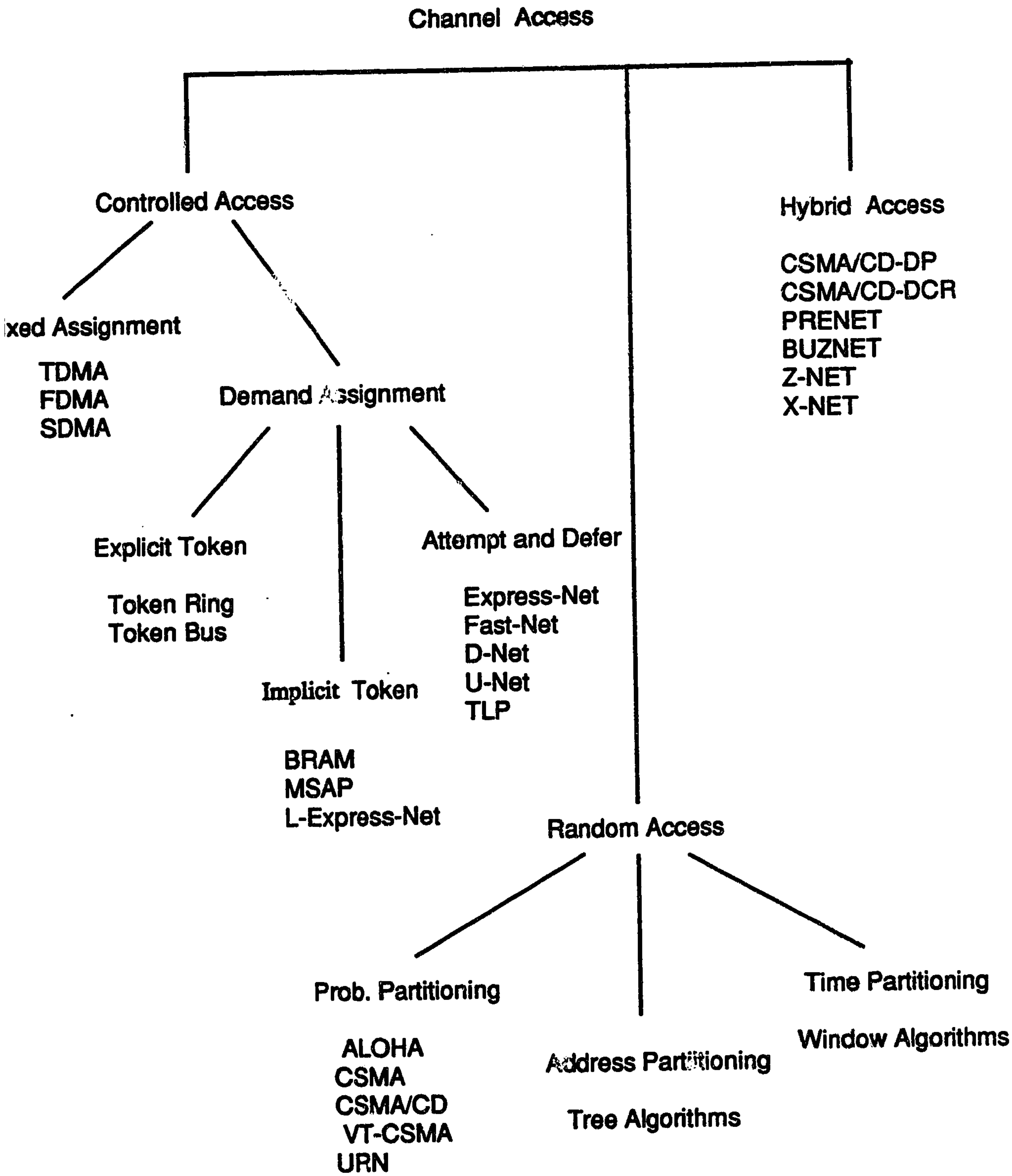


Figure 1.1: Multiple access protocols classes

Recognition Access Method) [7], MSAP (Mini-Slotted Alternating Priorities) [8] and L-Express-net [9]. In such protocols, the order in which channel access is granted is determined by a logical order of stations, which may or may not correspond to their physical order on the transmission medium.

Another demand assignment scheme is the *attempt and defer* mechanism which is suited for the unidirectional bus systems. In such a scheme, a station transmits if it finds the channel to be idle. However, if in the middle of its transmission, the station detects a transmission from an upstream station, it aborts its transmission, deferring to the upstream station. Access protocols employing the above scheme include protocols of Express-net [10], D-Net [11], Fasnet [12], U-net [13] and tokenless protocols [14].

1.1.2. Random Access Protocols

The second class of multiple access protocols is random access protocols. Most random access protocols employ two algorithms: a channel access algorithm and a collision resolution algorithm (CRA). The former algorithm is used to assign stations transmissions rights, while the latter is responsible for resolving collisions once they occur.

Random access protocols could be characterized according to the way stations are granted transmission rights. Three categories have been identified in [3]: probabilistic, address-based and time-based partitioning.

Probabilistic partitioning:

In probabilistic partitioning schemes, a station transmits its packet with some probability P or defers its transmission until some future time with probability 1-P.

An example of such schemes is the ALOHA protocol [15] in which stations are allowed to transmit whenever they wish without listening to the medium, i.e., P=1 (no partitioning). Improvements to the ALOHA protocol led to the evolution of a family of carrier sense multiple access protocols (CSMA) [16-18].

In the p-persistent CSMA [16], a station senses the channel before transmission. If the channel is sensed idle, the station transmits its packet with probability p or defers its transmission for a fixed amount of time (usually referred to as a slot) with probability 1-p, and reattempts transmission at that latter time. If, however, the channel is sensed busy, the packet transmission is rescheduledd until a later time.[1] If two or more stations decide to transmit on the channel at the same time[2], a collision occurs. All the packets involved in the collision are rescheduledd after the collision is detected.

CSMA/CD protocols save some of the bandwidth wasted in collisions by requiring stations to listen before and during packet transmissions, as opposed to just before transmission in the CSMA case. Detection of a collision causes a station involved in the collision to abort its packet transmission. Ethernet [17] employs a 1-persistent CSMA/CD protocol.

The reschedulingg process differs from one protocol to another. For instance, in ALOHA packet reschedulingg is taken from the same uniform distribution every time a collision occurs. In Ethernet, a probabilistic scheme called the *binary backoff* algorithm is used. The binary backoff scheme is similar to the one employed in ALOHA, except that

1. In 1-persistent CSMA, the station persists until the channel is idle.
2. "At the same time" means within a time interval which is referred to as the collision window.

the mean of the uniform distribution doubles with every collision.

Molle and Kleinrock [18] suggested VT-CSMA (Virtual Time CSMA) for multiple access communication. In VT-CSMA, each station has two clocks, a real time clock and a virtual time clock. The virtual clock stops running when the channel is busy, and runs when the channel is idle. When it runs, the virtual clock runs at a rate higher than the real one if it is behind the real clock. A packet is transmitted only when its arrival time is equal to the time on the virtual clock.

Another probabilistic collision resolution algorithm is the one used in the URN protocol [19]. Assuming that the multiplicity of a collision could be known and is given by n out of N stations, the Urn protocol enables a set of K stations such that the probability of exactly one station is ready (has a packet to send) is maximized. K is given by N/n.

Address partitioning:

The first address partitioning random access protocols were proposed independently by Capetanakis [20] and Hayes [21]. In their protocols time is slotted. Assuming that a number of stations become ready in an otherwise idle system, those stations attempt to transmit. If a collision occurs, stations are split into two halves, an enabled half and a disabled one. Stations of the enabled half are allowed to contend for the channel. If further collisions occur, the enabled set is continually halved until eventually no collisions occur. If at some point the enabled set had no ready packets, an empty slot occurs and the disabled set becomes enabled. Because of the binary nature of the protocol, it is often referred to as the Binary tree algorithm, and is similar to the binary tree search algorithm.

Massey [22] and Tsybakov and Mikhailov [23] independently proposed a modification to the basic tree algorithm above, known as the level-skipping tree algorithm. In the level skipping tree algorithm, if a slot containing a collision is followed by an empty slot, then, rather than have the users get involved in a sure collision, they are immediately halved into two subsets, thus saving one slot.

Recently Greenberg [24] and Cidon and Sidi [25] have shown that by estimating the conflict multiplicity before invoking the CRA, a smaller number of slots would be required to resolve a collision. Indeed, it is shown in [25] that higher throughputs are achievable by estimating the conflict multiplicity, before invoking the CRA.

Time partitioning:

Protocols in this group are known as window protocols, after Gallager's window protocol [26]. In [26], packet transmission is conducted in cycles, where the i^{th} transmission cycle is used to transmit packets that arrived in the i^{th} arrival interval (window). If Δ is the window size, then the i^{th} window is the interval $(i \cdot \Delta, i \cdot \Delta + \Delta]$. The splitting is based on arrival times of packets, by halving the window, and allowing the packets that arrived in the first half to be transmitted and so on. The resulting algorithm is, thus, a first-come-first-served (FCFS) algorithm.

Gallager [26] also observed that if a collision is followed immediately by another, then no new information is known about the other half of the interval corresponding to the former collision. Thus, those packets are returned to the unexamined portion of the arrival axis. By applying this observation, a higher throughput was achieved.

Mosely and Humblet [27] devised a window algorithm for the Poisson arrival model based on optimizing the window size at each step using dynamic programming techniques with the cost function being the throughput. Panwar et al [28] introduced a similar algorithm but for the Bernouilli arrival model. Several other variations of the window protocol exist in the literature.

1.2 Alternative Approaches

Random access protocols perform very well at light load. However, as was discussed above, some of the channel capacity is wasted on collisions, as well as collision resolution. On the other hand, controlled access protocols perform very well at heavy load (round robin fashion) but suffer unnecessary delays at light load, since stations have to wait until they acquire transmission rights.

Solutions to this problem fall into two categories: Hybrid access protocols and collision-avoidance switches.

1.2.1. Hybrid Access Protocols

This group of protocols are referred to as Hybrid since they interchangeably employ two modes of operation: a contention mode at light load and a controlled access one at heavy load. Examples of such protocols are CSMA/CD with dynamic priorities and low collision probability (CSMA/CD-DP) [29], CSMA/CD with deterministic collision resolution (CSMA/CD-DCR) [30], PRENET [31], Buzz-net [32], Z-net [33], X-net [34], and the protocols in [35] and [36].

All of the above protocols operate in a random access fashion at light load. At higher loads, some sort of controlled access scheme is employed. For instance, in [29] and [30] a deterministic collision resolution scheme is employed using a reservation method. In [29], however, a clustered station access scheme is used, thus reducing contention.

In PRENET [31], a pre-emptive scheme is used to avoid collisions, with the left-most station having the highest priority. (This, however, is performed under both light and heavy traffic conditions, i.e. the protocol will result in no collisions).

In Buzz-net [32] and Z-net [33], the controlled access mode is done in a round robin fashion starting from the same end station. A similar scheme is conducted in [34] except that the cycle begins from opposite ends each time (hence the name X-net). In [35], when collisions occur, stations operate in a round robin fashion, starting from the left-most station, with new arrivals allowed to be transmitted.

1.2.2. Collision-Avoidance Switches

This approach uses hardware collision avoidance switches, in a star shaped network, to prevent collision. The collision-avoidance circuit (a collection of single-bit arbiter circuits) is placed at the center of the star network. The purpose of the switches is to arbitrate random access to a common communication channel. The circuit would have N incoming links (corresponding to an N-node system) and a single common outcoming link. This link is hence referred to as the channel. The channel, then, branches in N different directions (corresponding to the number of nodes). These links are known as the outgoing links. While the channel is busy, the switches block traffic from all nodes, except the one currently transmitting.

Collision-Avoidance Star Local Area Networks (CASLANs), see Figure 1.2, were discovered independently by Lee and Boulton [37], Closs and Lee [38], and Albanese [39]. The access protocol for CASLANs is based on repeated attempts by nodes to acquire the hub (central node). The protocol consists of two components, one executed by the nodes, and the other by the hub.

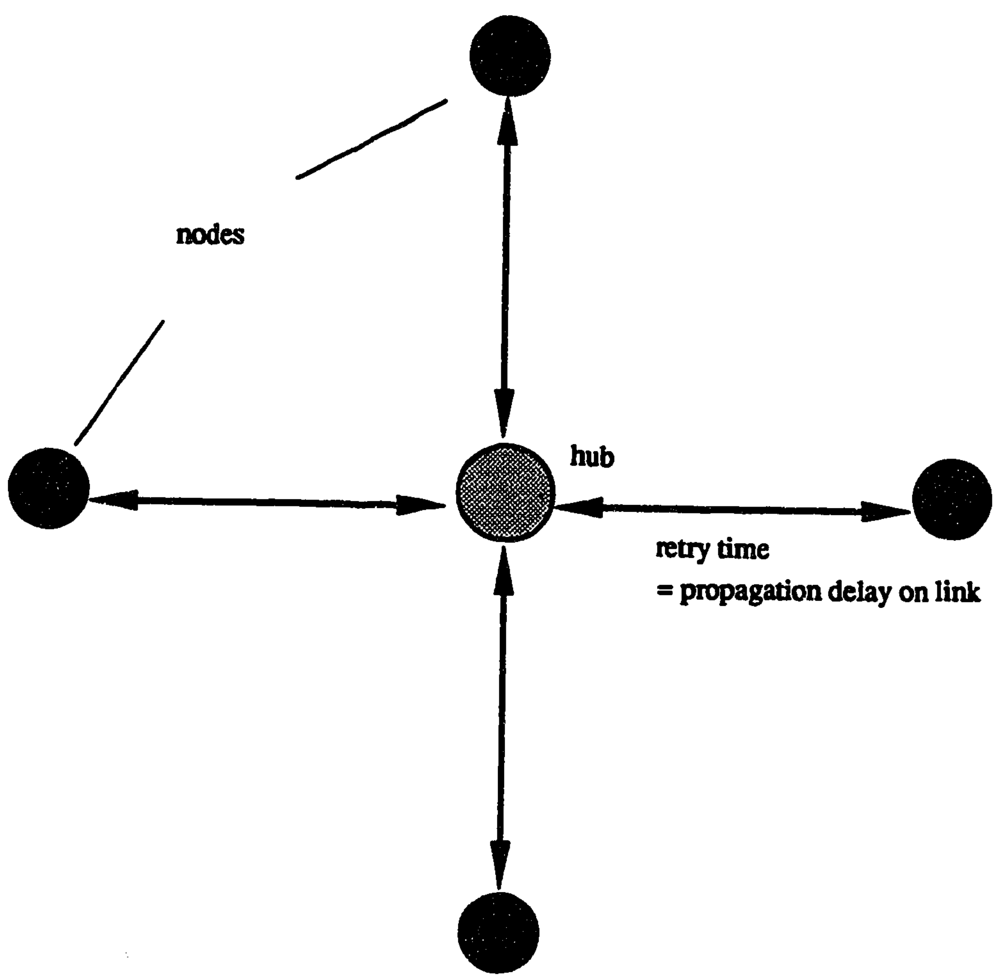


Figure 1.2 : The topology of CASLANs

Nodes are either idle (have no packets to transmit) or ready (have packets that have not yet acquired the hub, or are acquiring the hub). The node protocol is as follows. As soon as an idle node receives a packet, it forwards that packet to the hub. A packet transmitted by a node acquires the hub if the hub is free upon the packet arrival at the hub. Otherwise, the packet is blocked. If a node does not sense its own packet after the round trip propagation delay between the node and the hub, it retries transmission immediately until successful. (The round trip propagation delay between a node and the hub is thus called its retry time). A node acquiring the hub cannot transmit a new packet until the last bit of its on going transmission has propagated back to the node.

The hub protocol is as follows. If a packet arrives at the hub and the hub is free, the packet acquires the hub instantaneously. Once a Packet acquires the hub, the packet is transmitted on all outgoing links. If two or more packets arrive at the hub at the same time, one of them is chosen at random and the other packets are blocked. Thus, collisions never occur in such networks, and valuable bandwidth is saved.

1.3. Thesis Outline

This thesis is concerned with the study of the behavior and performance of CASLANs. Several mathematical, as well as simulation models are proposed to evaluate the performance of CASLANs.

Wherever possible exact performance models are sought. In many cases, however, such models are either mathematically intractable and/or computationally expensive. In such cases, assumptions and approximations are unavoidable. In cases where approximate analysis is used, the results of the system analysis must be complemented with those of an exact simulation model that does not bear any of the restricting assumptions of the mathematical model and, therefore, providing an accurate basis for evaluating the validity and the effect of the simplifying assumptions and approximations used in the mathematical analysis. In the event that a valid and useful mathematical model cannot be devised, performance evaluation would then rely on simulation models. The trouble with simulation models, however, is that they are expensive to develop, debug and run.

The performance measures sought are dependent on the applications that CASLANs can be used to support. For instance, in applications where only data packets are available, the network utilization and average packet delays are the performance measures to seek. Whereas, for real-time applications, e.g., voice traffic, the primary performance measure would be the percentage of packets received before their delivery deadlines expires. In such cases the packet delay distribution, and no longer its mean value, is the primary performance measure.

The network *utilization* is the fraction of time the channel is used. Another term commonly used to describe the use of the channel is the system *throughput*, which is the fraction of time the channel is used for the transmission of useful information, i.e., excluding all overhead bits. In the course of this thesis, both terms are used interchangeably to refer to the network utilization since one could be easily computed from another. The maximum achievable utilization of a network is often referedd to as its *capacity*. It is intuitively clear that the capacity of collision-free LANs, e.g., CASLANs, is generally higher than those that exhibit collisions.

The *packet delay*, also often referedd to as the *transfer delay*, is defined as the elapsed time from the instant that a message arrives at a node to its successful reception at its destination node. This delay includes several components, namely, queueing delay due to messages queued at their source nodes, access delay, which is the time a packet spends trying to access the channel, packet transmission time and propagation delay from source to destination. The term packet delay used in this thesis only includes the latter three components above. Since transfer delay is measurable by the user, e.g., interactive user applications, it is a more important performance measure in LANs than the network capacity. Therefore, in this thesis, more emphasis is put on average delay performance results.

As mentioned earlier, the focus of this thesis is on CASLANs. Such focus can be justified by the attractive performance and practical features of CASLANs:

(1) The node interconnection in CASLANs is point-to-point. That is, the network interface could be active, therefore, making CASLANs very well suited for use in optical fiber communication [37].

(2) CASLANs were shown to be workable at very high transmission speeds, e.g., 100 Mbps, without any degradation in performance.

(3) The capacity of CASLANs is very high. It was shown [45] that CASLANs can operate at a 98% utilization. Moreover, it was shown [44, 45, and 47] that such a high a capacity is achievable while maintaining very reasonable average delay values.

(4) The simplicity of the hub and node protocols. As mentioned earlier, the node protocol is simply based on repeated attempts to acquire the hub and the hub role is simply to arbitrate between the different nodes.

In studying CASLANs, research in this thesis branches in three directions:

(1) The first is an indepth study of the performance and behavior of CASLANs. The effect of several network and traffic parameters are studied, most visible of which is the effect of the retry time. This part is the topic of chapters 2 and 3.

(2) The second involves an investigation of the performance of CASLANs in a time-constrained environment. Both soft and hard real-time applications are considered. This part is presented in chapter 4; and

(3) The third involves studying and analyzing the performance of prioritized CASLANs. Both data only and integrated voice and data applications are considered. This is contained in chapter 5.

Chapter 2 starts with a simulation study of CASLANs exploring the effect of node retry time values on average performance measures. An inherent and interesting relation between the packet length and retry time, for fixed-packet-length CASLANs, is discovered.

It is shown that, depending on network parameters, increasing the retry time does not always result in an increased average packet delay. Indeed, at retry time values that are factors[3] of the packet length a local minima is observed for average delay results[4]. Also, it is

3. A retry time value τ is a factor of a packet length T if $T \bmod \tau = 0$.

shown that, in contrast to what was believed earlier, packets transmitted with a retry time that is a factor of the packet length are guaranteed transmission rights. A heavy load analysis of CASLANs is presented. The analysis shows that, under heavy load conditions, packet transmission in CASLANs is round robin and that the delay is bounded.

In view of the performance results presented in chapter 2, a new CASLANs performance model that is capable of capturing the inherent relation between the packet length and the retry time, becomes a necessity. In chapter 3, such a model is introduced.

The model is based on following a tagged user in an exact analysis and approximating the behavior of the rest of the users by that of the tagged user at steady state using an iterative approach. A polling system is used to represent CASLANs where the hub chooses at random one of the idle or ready users. Results show that the model is very accurate compared to simulation results, and is more accurate than all earlier CASLANs models [39-45].

Chapter 4 deals with CASLANs in a real-time environment. For soft real time systems, a simulation model is developed from which the packet survival functions are extracted and subsequently the probability of packet loss for any desired deadline. For hard real-time systems, a performance model for exponentially distributed packet laxities is introduced. In this model, packets that exceed their laxities are removed from the system. This model is an extension of the model in chapter 3 with the added complexity of the possibility of packet loss. Results of the model show that it is an accurate representation of CASLANs in a hard real-time environment. Generally speaking, the work in this chapter shows that

4. That is, they show some form of performance superiority.

CASLANs perform very well under real-time constraints. The results also show that factor retry time values seem to extend their superiority to real-time applications.

In chapter 5, a study of prioritized CASLANs is presented. The chapter begins by the analysis of the behavior of a CASLAN with two priority classes. Retry times are used for priority assignment. It is, again, shown that a lower retry time does not always mean higher priority. Indeed, and as in the single class case, a class with a retry time that is a factor of the packet length could have a higher priority than a class of users with a lower, non-factor, retry time value. Because of the extreme complexity and computational expense of a two-class CASLANs model that follows tagged users exactly is not sought. However, another simpler more restrictive model is presented. The model assumes independence between a packet retry attempts at the hub. Such a model, however, proved to be accurate only for moderately loaded system. In the final part of the chapter, a study of integrated voice and data applications on CASLANs is presented. The study is based on a simulation model. Results show that, even for large systems and under heavy data load conditions, CASLANs network parameters could be set such that no packet loss is experienced.

Chapter 2

Behavioral Analysis of CASLANs

The purpose of this chapter is to explore the effect of node retry time values on the performance and behavior of CASLANs. Such an effect has, indeed, been left out in previous CASLANs performance studies [37-48]. The general consensus in all these studies is that as the retry time increases so does the delay. In this chapter, it is shown that the above belief is not true. That is, the average packet delay does not necessarily increase with the retry time. An inherent relation between the retry time and the packet length is discovered (section 2.1) and explained (section 2.2). Also, an exact analysis of CASLANs, under heavy load conditions, is presented (section 2.3).

2.1 Simulation Results

In this section, a simulation model for studying and analyzing the performance of symmetric CASLANs is introduced. By symmetric it is meant that all nodes have the same packet generation process, retry time and mean packet length.

The simulation model incorporates all the features of the CASLANs protocol under the following conditions:

1- All transmitting nodes are equipped with single buffers. That is, once a packet occupies the buffer, no new packets are generated by a node unless the packet currently in the buffer has been successfully transmitted.

2- No packet loss due to network failure or buffer overflow at the receiving end is considered.

3- The transmission medium is noiseless and error free.

4- Two packet length characteristics are considered: variable-length and fixed length packets. Variable-length packets were chosen to be exponentially distributed. Such a choice was merely done to study the effect of the retry time on variable-packet-length CASLANs. It is assumed that conclusions drawn under this assumption generally hold for other distributions (but not for the deterministic one).

5- The round trip propagation delay between any node and the hub is the same for all nodes, and is equal to the retry time.

6- The delay of a packet is measured from the time the packet is generated until the time of the complete packet reception by the destination.

The rationale behind using the single buffer assumption is that the emphasis of this thesis is on the channel access and transmission delays and not on queueing delays. Since such an assumption does not affect performance results of channel access and transmission delays, it is used to simplify the simulation model.

In the simulation runs in this section, and throughout the thesis, 95% confidence intervals were obtained. Since the confidence intervals were always too small, they are not shown in the plots. The number of packets transmitted during a simulation experiment was chosen such that a stable performance of the system is reached.[5]

5. This number was usually 50,000 packets.

Let N, T, λ, and τ be the number of nodes, average packet length, packet arrival rate per time unit and retry time of the CASLAN to be simulated. Let the term *offered load* represent the number of packets arriving during an average packet transmission time. The offered load is, then, given by $N \times T \times \lambda$. Because of the single buffer assumption, high to heavy carried load (system utilization) may require offered load values greater than 1, e.g., offered load=9.6 in Figures 2.1 and 2.2. In the examples to follow, and in fact in the remainder of this thesis, the bit transmission rate is chosen to be 50 Mbps. Such a bit transmission rate was chosen since it is used in Hubnet, a 50 Mbps commercial CASLAN. In the results in Figures 2.1 and 2.2, N is chosen to be 20 nodes. Even though such a number may seem smaller than that in many practical systems, it is the system behavior that we are interested in at this stage. Indeed, in chapter 5, CASLANs with more than 1000 nodes are examined. The average packet length was randomly chosento be 480 bits. Such a choice, however, is irrelevant because the interest in this chapter is to study the effect of the retry time on the performance of CASLANs. As will be shown, changing the retry time while fixing the packet length should be equivalent to varying the packet length while fixing the retry time.

Figure 2.1 shows the average delay versus the retry time of a symmetric CASLAN (with $N = 20$ nodes and a variable packet length with mean 480 bits) at different values of the packet arrival rate per node, λ. As expected, and as has been reported in previous studies, as the retry time increases the average delay increases.

Figure 2.2 shows the average delay versus the retry time for a symmetric CASLAN with N=20, with a fixed packet length (T = 480) at different values of λ. At light load

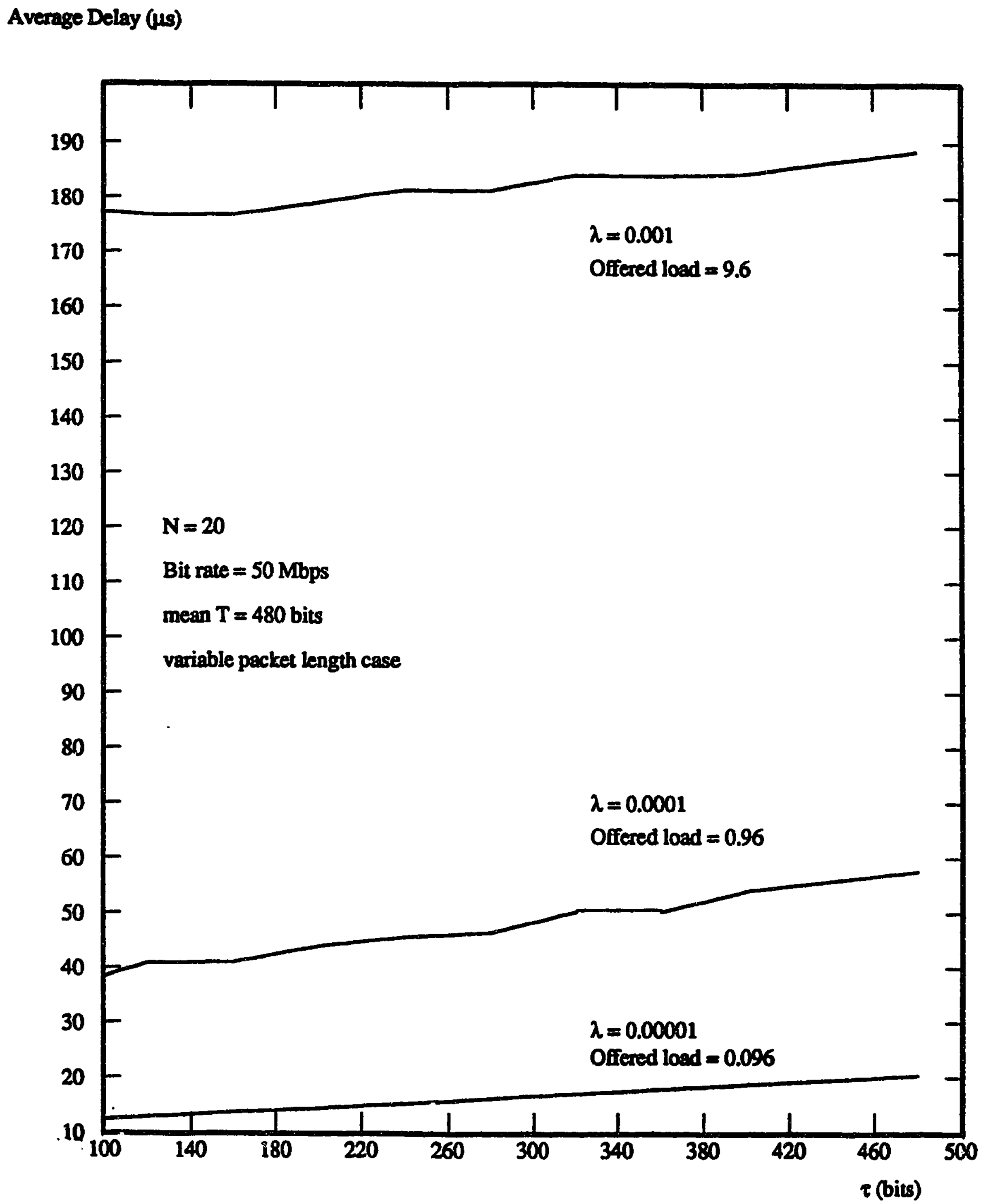


Figure 2.1: Response time at different values of τ in a 20 node system

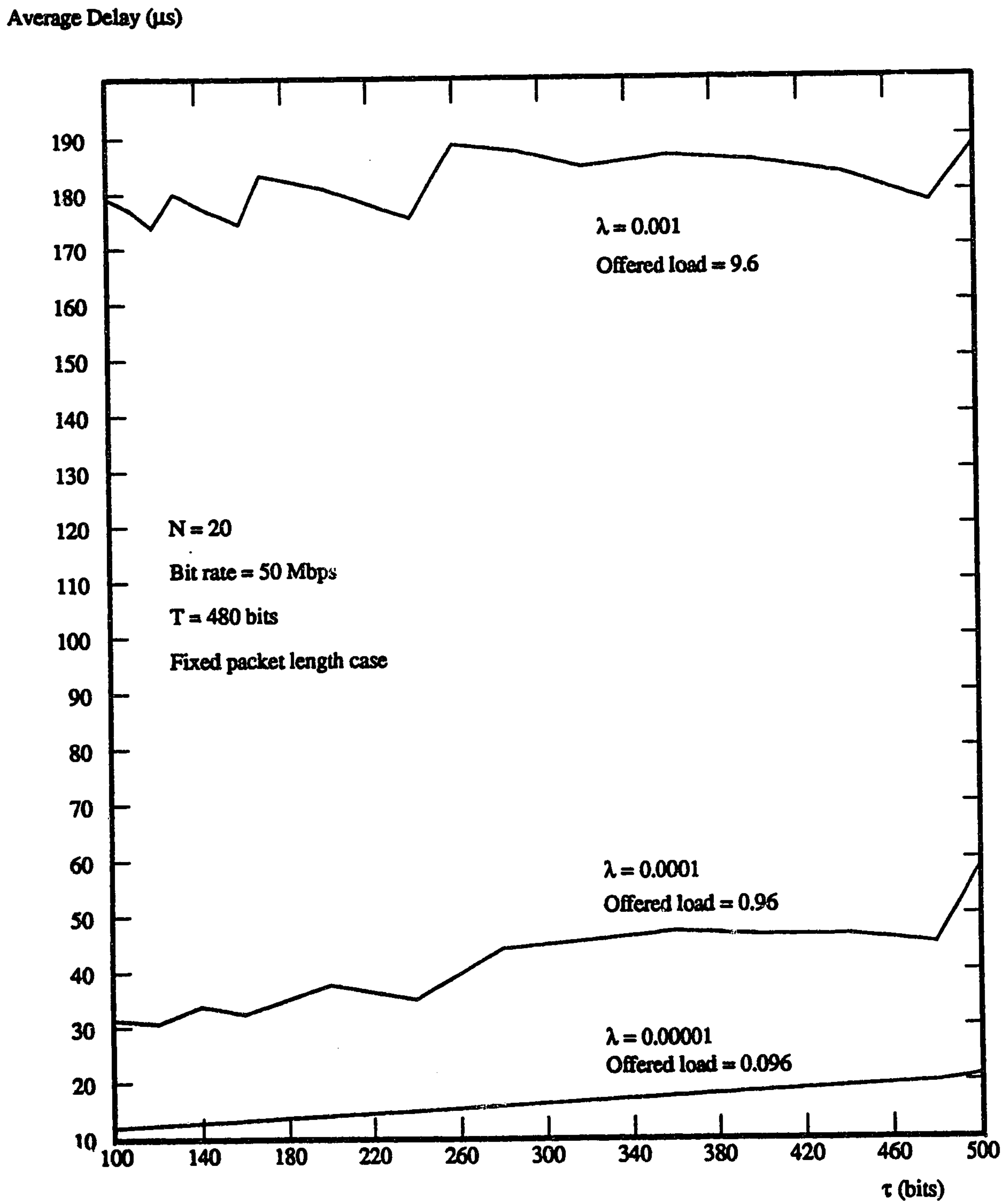


Figure 2.2: Response time at different values of τ in a 20 node system

(e.g., $\lambda = 10^{-5}$, offered load = 0.096) the greater the retry time, the greater the delay. This result does not always hold at higher loads. For example, at $\lambda = 0.001$ (offered load = 9.6), the average delay at $\tau = 100$ (bits) is 179 μs, whereas at $\tau = 120$ the average delay is 173μs. That is, the delay at a higher value of retry time (120) is less than the one at a lower retry time (100). This result, which defies intuition at first glance, is due to the fact that 480 (packet length) is a multiple of 120 (retry time). The same phenomenon is observed for every τ which is a factor of T (such as 160, 240 and 480 bits). At all of these τ values, a drop in the mean packet delay is observed, which results in a local minimum of the average delay at these values. Similar results were observed for other values of the number of nodes and packet length values.

In the remainder of this chapter, the case where the retry time is less than or equal to the packet length is emphasized. To show why such a case is a practical one, consider the following example. For an instance of CASLANs, viz. Hubnet [37], where the minimum packet length is 88 bits, and with a bit rate of 50 Mbps, it means that we are considering a network with a diameter of about 530 meters (for local networks distributed within a building or some adjacent buildings, this distance seems very reasonable). The study is confined to the case where the packet length is fixed. This is due to the following

(1) Many applications, such as voice communication and uncompressed video communication employ fixed-packet-length transmission.

(2) For the variable packet length case, the average packet delay increases with the retry time. Therefore, studying the effect of the retry time on the performance in

such a case does not provide any new insight into the behavior of the network.

We begin our study of the effect of the retry time on the performance of CASLANs, by making the following observation from the simulation results.

Proposition 2.I:

For the fixed-packet length (T) symmetric CASLANs, for every τ_f that is a factor of T, there exists a period $[\tau_f - \delta, \infty)$ in which $D(\tau_f) \leq D(\tau)$, where $\tau \in [\tau_f - \delta, \infty)$ and D(X) denotes the average delay at retry time X. The period $[\tau_f - \delta, \infty)$ is called the *range* of τ_f.

□

That is to say, if the retry time is a factor of T, then the mean packet delay at that retry time is better than every higher retry time and some lower retry times subject to its range. This proposition is based on observations from simulation results. We do not provide a proof for it nor do we provide a general expression for the range of a factor retry time value. However, an expression for the range is provided for the heavy load case (see section 2.4).

The width of that range, and in particular δ, is dependent on the load and the number of nodes. For instance, at very light load $\delta = 0$ (see Figure 2.2, at $\lambda = 10^{-5}$), since the delay always increases with τ. At heavy load, however, $\delta > 0$. For instance, at $\lambda = 0.001$ (offered load = 9.6), the range of 240 is approximately $[170, \infty)$.

A direct consequence of proposition 2.I is that if τ_1 and τ_2 are two factors of T and $\tau_1 < \tau_2$, then $D(\tau_1) \leq D(\tau_2)$. This follows from the non-existence of an upper bound on the range of τ_1. This fact is confirmed by the results in Figure 2.2, $D(120) \leq D(160) \leq D(240)$, and is proven in the following section.

In view of the above results, it is appropriate that the following modes of operation of CASLANs are studied.

1) CASLANs with retry time that is a factor of the packet length. The objective here is to find why such retry times are special.

2) Since the above phenomenon is more apparent at higher loads, a heavy load analysis of CASLANs is conducted.

2.2. CASLANs with a Retry Time That is a Factor of Packet Length

In this section, the behavior of CASLANs in which the retry time is a factor of the packet length is studied and analyzed. Let τ (the retry time) be a factor of T (the packet length) such that T is constant, and $T = K \cdot \tau$, where $K > 0$ is an integer number.

We begin by making the following definitions. Recall that the CASLANs protocol is based on repeated attempts by the nodes to acquire the hub. Assume that a node makes two successive attempts to acquire the hub, and both attempts are unsuccessful. Then if the second one arrives at the hub during the same hub acquisition period as the first one, the second one is called an *intrapacket* arrival. Otherwise, i.e., if it arrives during a new acquisition period, it is called an *interpacket*. Therefore, for any node, and during any hub acquisition period, there is exactly one interpacket arrival and zero or more intrapacket arrivals. The remainder of the hub acquisition time on the l^{th} interpacket arrival of a certain packet is denoted by X_l, and the time from the start of the hub acquisition by Z_l. The latter is called the l^{th} interpacket arrival time. Obviously, $Z_l = T - X_l$. The remainder of the hub acquisition time at the point of the last intrapacket arrival is denoted by Y_l. The overhead period between hub acquisitions is denoted by t, see Figure 2.3. It should be noted that the values of X_1 and Z_1 are not very meaningful. This is because a packet may arrive randomly during the hub acquisition on its first interpacket arrival, see X_1 in Figure 2.4.

Theorem 2.1:

If X_{l-1}, X_l, Z_{l-1} and Z_l are as defined above and τ is a factor of T, then for every packet being transmitted

$$1)\ X_l \geq X_{l-1}$$

and

$$2)\ Z_l \leq Z_{l-1}$$

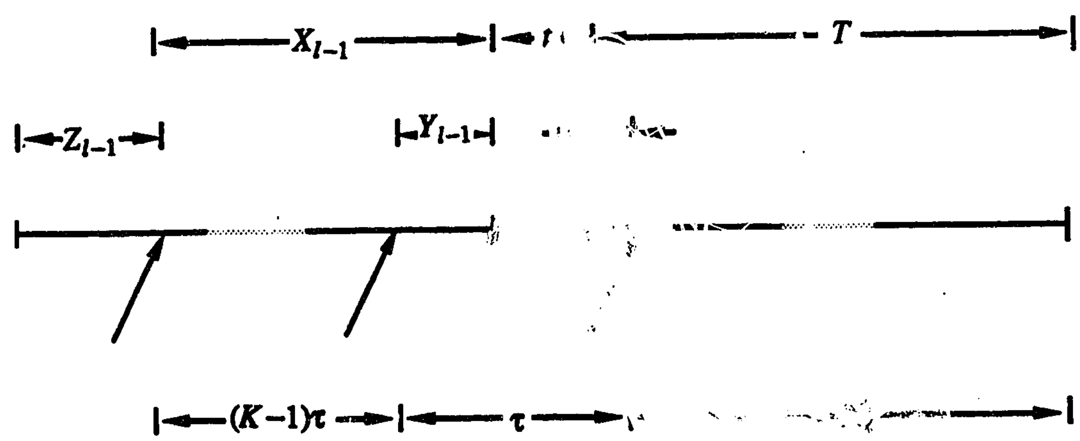


Figure 2.3: Relation between the $l-1^{st}$ and the l^{th} interpacket arrivals of a packet

Proof:

1) From Figure 2.3, with $T = K \cdot \tau$, it is easy to see that

$$X_l = T + t + Y_{l-1} - \tau$$

But

$$Y_{l-1} = X_{l-1} - (K-1)\cdot\tau$$

Hence,

$$X_l = T + t + X_{l-1} - (K-1)\cdot\tau - \tau$$

$$= T + t + X_{l-1} - K\tau$$

$$= t + X_{l-1}$$

Since t, the overhead between transmissions is ≥ 0, then

$$X_l \geq X_{l-1}$$

2) Since $Z_l = T - X_l$, then it follows from part (1) that $Z_l \leq Z_{l-1}$.

□

The above theorem implies that the interpacket arrival time of a transmitted packet keeps getting smaller[6] with the number of interpacket retry attempts. An interpacket arrival time of $Z_l = 0$ means that the packet has acquired the hub and will be transmitted.

Using theorem 2.1 above, it is next proven that if τ_1 and τ_2 are both factors of T, and if $\tau_1 < \tau_2$, then the average value of the second interpacket arrival time in the case of τ_1 is no more than the average value of the second interpacket arrival time in the case of τ_2. Let

6. An exception would be the case where the hub is never idle. In this case the interpacket arrival does not decrease and will never reach 0 (successful transmission). Such a situation, however, is almost practically impossible since it implies that the processing time at the hub is always zero, and that packet transmissions are synchronized.

$X_l(\tau)$, $Y_l(\tau)$ and $Z_l(\tau)$ denote the values of X_l, Y_l and Z_l in a CASLAN with retry time τ, respectively.

Theorem 2.2:

Let τ_1 and τ_2 be both factors of T. If $\tau_1 < \tau_2$, then

$$E[Z_2(\tau_1)] \le E[Z_2(\tau_2)],$$

if the following hold:

1- The packet generation process is Poisson.

2- The length of the overhead period, t, has the same distribution whether the retry time of the tagged user is τ_1 or τ_2.[7]

Proof:

Let W denote the time of arrival of a tagged packet on its second interpacket arrival, measured from the last hub release. Thus, $W = Z_2 + t$, see Figure 2.4.

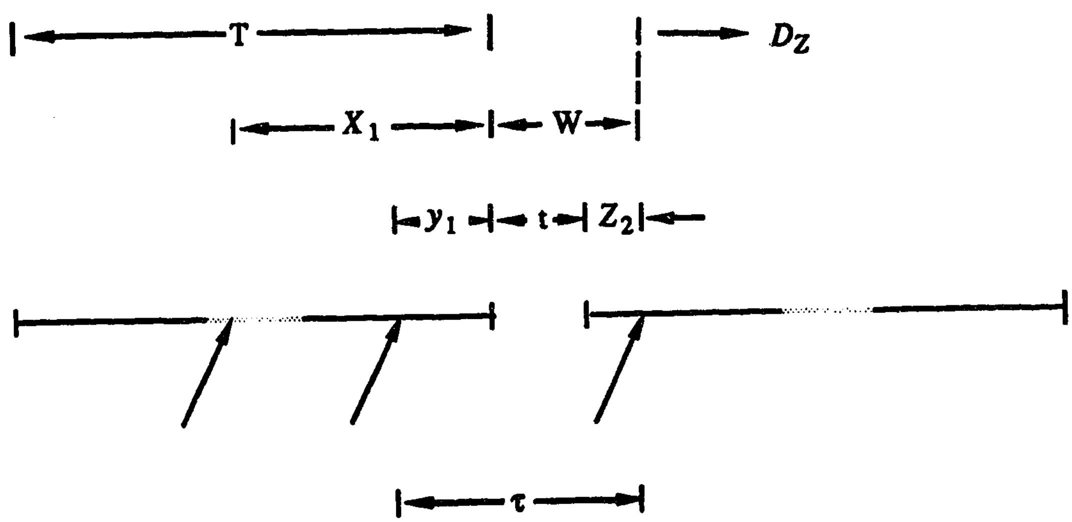


Figure 2.4 : Relation between X_1, Z_2 and W

7. We assume that the tagged users behavior is distinct from that of other users.

Let $f_W(u,\tau)$ denote the pdf of W in a system with retry time τ, and is given by

$$f_W(u,\tau)=\sum_{i=1}^{k} f_{X_1}(i\tau-u) \tag{2.1}$$

where, $f_{X_1}(v)$ is the pdf of the remainder of the hub acquisition time upon a packet arrival from an idle node and is given by

$$f_{X_1}(v)=\frac{\lambda e^{-\lambda(T-v)}}{1-e^{-\lambda T}} \qquad 0<v\le T \tag{2.2}$$

Therefore,

$$f_W(u,\tau)=\sum_{i=1}^{k}\frac{\lambda e^{-\lambda(T-i\tau+u)}}{1-e^{-\lambda T}}$$

$$=\frac{\lambda e^{-\lambda u}}{1-e^{-\lambda\tau}} \tag{2.3}$$

Now, let $W(\tau)$ and $t(\tau)$ respectively denote the values of W and t in a system with retry time τ. Hence,

$$E[W(\tau)]=\int_0^{\tau} u\cdot f_W(u,\tau)\cdot du$$

$$=\int_0^{\tau} u\cdot\frac{\lambda e^{-\lambda u}}{1-e^{-\lambda\tau}}\cdot du$$

$$=\frac{1}{\lambda}-\frac{\tau}{e^{\lambda\tau}-1} \tag{2.4}$$

Now if $\tau_1<\tau_2$, then it is not hard to see that

$$\frac{\tau_1}{e^{\lambda\tau_1}-1} \geq \frac{\tau_2}{e^{\lambda\tau_2}-1}. \tag{2.5}$$

From (2.4) and (2.5)

$$E[W(\tau_1)] = \frac{1}{\lambda} - \frac{\tau_1}{e^{\lambda\tau_1}-1} \leq \frac{1}{\lambda} - \frac{\tau_2}{e^{\lambda\tau_2}-1} = E[W(\tau_2)]. \tag{2.6}$$

Since

$$W(\tau) = t(\tau) + Z_2(\tau),$$

then

$$E[W(\tau)] = E[t(\tau)] + E[Z_2(\tau)] \tag{2.7}$$

Substituting in (2.6) above, then

$$E[t(\tau_1)] + E[Z_2(\tau_1)] \leq E[t(\tau_2)] + E[Z_2(\tau_2)]$$

Since, according to assumption 2, the overhead period, t, has the same distribution with retry times τ_1 and τ_2, then $E[t(\tau_1)] = E[t(\tau_2)]$. Thus

$$E[Z_2(\tau_1)] \leq E[Z_2(\tau_2)]$$

□

Let D_Z denote the remainder of the packet delay at the hub starting from the second interpacket arrival of a tagged packet (see Figure 2.4). In the following, we show that $E[D_Z \mid Z_2]$ is proportional to the value of Z_2.

Theorem 2.3:

$E[D_Z \mid Z_2]$ increases with the value of Z_2.

Proof:

Let n be the number of interpacket transmissions before a user acquires the channel. That is, the packet acquires the channel on its $(n+1)^{st}$ interpacket arrival, i.e. $Z_{n+1}=0$.

$$E[D_Z \mid Z_2]=\sum_{n=2}^{\infty}\int_{t_n}\cdots\int_{t_2}(nT-Z_2+u_2+u_3+\cdots+u_n)\times$$

$$f_{(t_2,t_3,\cdots \mid t_n)}(u_2,u_3,\cdots u_n \mid n)\cdot d_{u_2}\cdots d_{u_n} \tag{2.8}$$

Where, $f_{(t_2,t_3,\cdots \mid t_n)}(u_2,u_3,\cdots u_n \mid n)$ is the probability of having overhead periods of length $t_2,t_3,\ldots,t_{n_r}$ given n_r interpacket attempts by the tagged node. Now, recall from theorem 2.1 that $Z_{l-1}=Z_l+t_{l-1}$. That is, $Z_2=Z_3+t_2$. Using this relation recursively, then

$$Z_2=Z_{n+1}+\sum_{i=2}^{n}t_i=\sum_{i=2}^{n}t_i \tag{2.9}$$

Substituting in (2.8), we have

$$E[D_Z \mid Z_2]=\sum_{n=2}^{\infty}\int_{t_n}\cdots\int_{t_2}(nT-Z_2+Z_2)\times f_{t_2,t_3,\cdots,t_n}(u_2,u_3,\cdots u_n;n)\cdot d_{u_2}\cdots d_{u_n}$$

$$=T\cdot\sum_{n=2}^{\infty}\int_{t_n}\cdots\int_{t_2}n\times f_{t_2,t_3,\cdots,t_n}(u_2,u_3,\cdots u_n;n)\cdot d_{u_2}\cdots d_{u_n}$$

$$=T\cdot E[n \mid Z_2] \tag{2.10}$$

From equation (2.9) above, it is not hard to see that $E[n \mid Z_2]$ increases with Z_2. Hence $E[D_Z \mid Z_2]$ increases with Z_2.

□

An interesting observation here is that the above result is independent of the retry time. That is, if τ_1 and τ_2 are both factors of T and $Z_2(\tau_1)\le Z_2(\tau_2)$, then $D_Z(\tau_1)\le D_Z(\tau_2)$.

Lemma 2.1:

Let τ_1 and τ_2 be both factors of T. If $\tau_1 < \tau_2$, and if the overhead periods are statistically independent of each other, then

$$E[D_Z(\tau_1)] \leq E[D_Z(\tau_2)]$$

Proof:

Let $\bar{t}$ be the average length of an overhead period. Thus

$$E[n \mid Z_2] = \frac{Z_2}{\bar{t}},$$

that is

$$E[n] = \frac{E[Z_2]}{\bar{t}}. \tag{2.11}$$

Now, consider equation (2.10) above. By unconditionning on Z_2, we get

$$E[D_Z] = T \cdot E[n]$$

$$= T \cdot \frac{E[Z_2]}{\bar{t}}.$$

Thus,

$$E[D_Z(\tau)] = T \cdot \frac{E[Z_2(\tau)]}{\bar{t}}.$$

From theorem 2.2, we know that if $\tau_1 < \tau_2$ and τ_1 and τ_2 are both factors of T, then

$$E[Z_2(\tau_1)] \leq E[Z_2(\tau_2)]$$

Therefore,

$$E[D_Z(\tau_1)] \leq E[D_Z(\tau_2)]$$

□

Now, $$D(\tau) = E[X_1(\tau)] + E[W(\tau)] + E[D_Z(\tau)] + T + \tau \qquad (2.12)$$

If it is assumed that $E[X_1(\tau)]$ is independent of τ, then from Theorem 2.2 and Lemma 2.1 above, for τ_1 and τ_2 both factors of T, and $\tau_1 < \tau_2$, $D(\tau_1) < D(\tau_2)$. That is, theorems 2.2 and 2.3 along with lemma 2.1 above imply that the range of a factor retry time value cannot extend to include the greatest lower factor retry time. For instance, referring to Figure 2.2, it can be noticed that

$$D(120) \le D(160) \le D(240) \le D(480),$$

where $D(\tau)$ is the average delay at retry time (τ).

The following theorem proves that if τ is a factor of T, then every packet transmitted over the channel will eventually be successfully transmitted. That is, no packet could face a busy hub indefinitely.

Theorem 2.4:

Let τ be a factor of T and let Q be a packet transmitted over the channel using τ. Q will acquire the hub with probability 1.

Proof[8]*:*

Let ε be the smallest time unit distinguishable by the hub, such that ε is very close to zero. Note that the range of X_l is $(0, T]$. Let the range of X_l be divided into m mini-periods each of length ε. It can be seen that the remainder of a packet transmission time evolves according to the Markovian property on a state space consisting of integer

8. The proof of this theorem uses a discrete, yet infinitesimal approach. The continuous case, however, can be treated by considering the limit as $m \to \infty$.

numbers between 0 and m. Note that $X_l = m$ corresponds to a sure acquisition of the hub. Let $P_{i,j}$ represent the transition probability from state i to state j (this means that $X_l - X_{l-1} = (j-i)\cdot\varepsilon$).

Using Theorem 2.1, it can be concluded that $P_{i,j} = 0$ if $j < i$. Thus,

$$\sum_{j=i}^{m} P_{i,j} = 1$$

Next, it is shown that state m is an absorbing state and that the probability that leaving any state i, $i<m$, state m is eventually reached is equal to 1. Mathematical induction is employed in the proof.

The first passage probability to state m given that the system was in state i, $f_{i,m}$, is given by

$$f_{i,m} = P_{i,m} + \sum_{j=i}^{m-1} P_{i,j} \cdot f_{j,m}. \tag{2.13}$$

Now

$$f_{m-1,m} = P_{m-1,m} + P_{m-1,m-1} \cdot f_{m-1,m}$$

$$= \frac{P_{m-1,m}}{1 - P_{m-1,m-1}} = 1 \tag{2.14}$$

Let us assume that $f_{j,m} = 1$, for $j \geq k$. Then,

$$f_{k-1,m} = P_{k-1,m} + \sum_{j=k-1}^{m-1} P_{k-1,j} \cdot f_{j,m}$$

$$= \frac{P_{k-1,m} + \sum_{j=k}^{m-1} P_{k-1,j} \cdot f_{j,m}}{1 - P_{k-1,k-1}}$$

$$= \frac{\sum_{j=k}^{m} P_{k-1,j}}{1 - P_{k-1,k-1}} = 1 \tag{2.15}$$

From (2.14) and (2.15) above, the theorem follows.

□

Thus $f_{i,m} = 1$, for $i = 0, 1, ..., m$-1. Since state m represents a successful transmission, this proves that, regardless of the point of arrival of a packet at the hub, it is guaranteed successful transmission if τ is a factor of T.

Even though it is conjectured that a retry time value that is not a factor of the packet length does not necessarily guarantee successful transmission for packets, a formal proof is not provided. Such a proof would require knowledge of the distribution of packet arrivals and the offered load. However, the following lemma is introduced and proven, to help in supporting such a claim.

Lemma 2.2:

Let T and τ be the packet length and retry time of a packet transmitted in a CASLAN, such that $T = k\tau + \Delta$, where $k \geq 1$ is an integer number and $0 < \Delta < \tau$. For any $l \geq 2$, there exists an infinite sequence of non-zero overhead periods $\{t_{l-1+j}, j=1 \cdots \infty\}$, at the hub, such that $Z_{l+j} = Z_l$, for $j = 1 \cdots \infty$.

Proof:

Consider the case where $Z_{l-1} \le \Delta$. From Figure 2.5a, we see that

$$T - Z_{l-1} + t_{l-1} + Z_l = (k+1)\tau \tag{2.16}$$

For $Z_l = Z_{l-1}$, then

$$t_{l-1} = (k+1)\tau - T = \tau - \Delta$$

Similarly by substituting $t_{l-1+j} = \tau - \Delta$ for $j = 1 \cdots \infty$, then $Z_{l+j} = Z_l$ for any $j \ge 1$.

Now consider the case where $Z_{l-1} > \Delta$. From Figure 2.5b, we see that

$$T - Z_{l-1} + t_{l-1} + Z_l = k\tau \tag{2.17}$$

By substituting $T = k\tau + \Delta$ and rearranging the terms, then

$$Z_l = Z_{l-1} - \Delta - t_{l-1}$$

If Z_l is chosen such that $Z_l \le \Delta$, then, as above, a sequence of $t_{l-1+j} = \tau - \Delta$, for $j = 1 \cdots \infty$, would make $Z_{l+j} = Z_l$ for any $j \ge 1$. Thus it only suffices to prove that there exists a value for t_{l-1} that makes $Z_l \le \Delta$.

For $0 < Z_l \le \Delta$, then

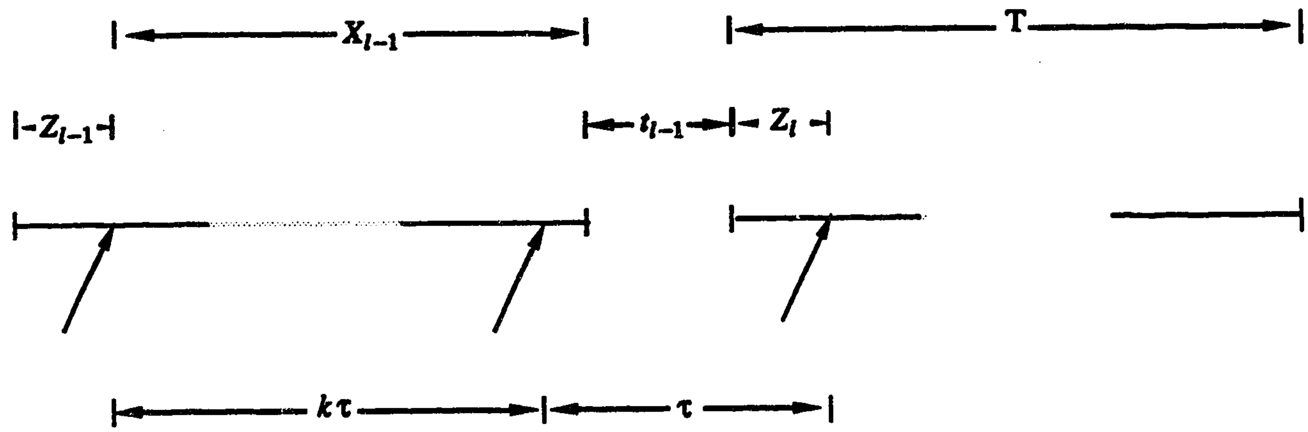
$$0 < Z_{l-1} - \Delta - t_{l-1} \le \Delta \tag{2.18}$$

Therefore,

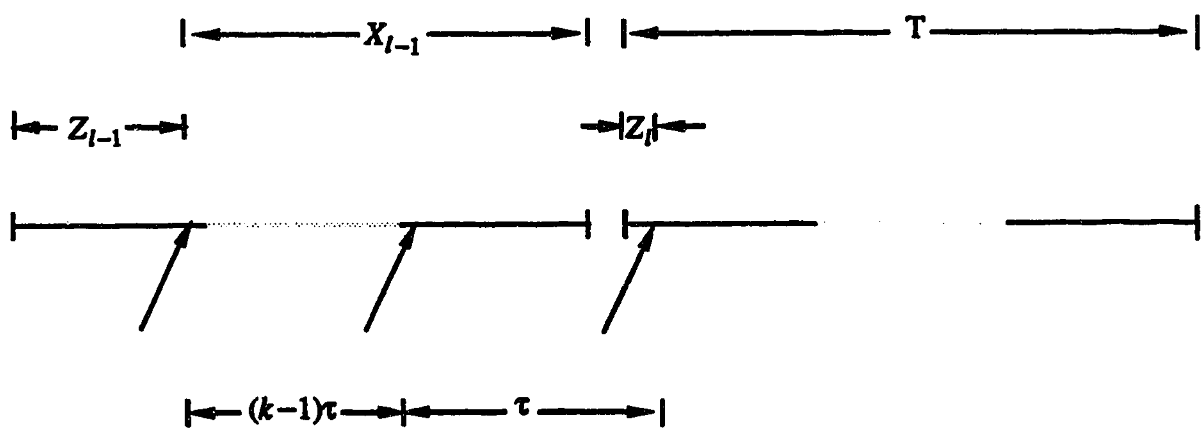
$$Z_{l-1} - 2\Delta \le t_{l-1} < Z_{l-1} - \Delta$$

Thus, $Z_l \le \Delta$ if t_{l-1} is chosen such that $t_{l-1} \in [\max(0,\ Z_{l-1} - 2\Delta),\ Z_{l-1} - \Delta]$. This completes the proof.

□



(a) $Z_{l-1} \le \Delta$, $t_{l-1} = \tau - \Delta$, i.e. $Z_l = Z_{l-1}$.



(b) $Z_{l-1} > \Delta$, t_{l-1} chosen such that $Z_l \le \Delta$.

Figure 2.5 : Relation between Z_{l-1} and Z_l for τ a non-factor of T.

It should be noted that the sequence of overhead values, in the above lemma, is by no means unique. However, it suffices to show that any such sequence exists to prove the lemma. It should be also noted, for retry time values that are factors of the packet length, no such sequence exists. (This follows from theorem 2.1.)

Lemma 2.2 proved the possibility of a never decreasing sequence of interpacket arrival times, for packets that are transmitted with a non-factor retry time value. Under such a situation, packets may never be successfully transmitted. Thus, it is appropriate to suggest the following proposition.

Proposition 2.II:

In CASLANs, a packet transmission with a retry time that is not a factor of the packet length may never acquire the hub. That is, such a packet is not guaranteed successful transmission.

□

Before concluding this section, the following comments are emphasized.

A retry time value that is a factor of the packet length was shown to provide some superior performance results. Indeed, not only does a retry time that is a factor of the packet length provide packets with always decreasing interpacket arrival times, but it also guarantees these packets an eventual transmission. Such advantages were reflected in the simulation results provided earlier, see section 2.1.

The relations and facts proved in this section apply regardless of the retry times with which other packets in the system are transmitted by. In fact, in chapter 5, it is shown that improved performance, of retry time values that are factors of the packet length, holds for the two-class and subsequently for the multi-class CASLANs.

2.3. Heavy Load Analysis

In this section, an exact analysis of fixed-packet length CASLANs operating under heavy load is presented. It is shown that if the system always operates under extremely heavy load, the delay is bounded. By extremely heavy load it is meant that the interarrival time between packets is very small and practically zero. That is, as soon as a packet is successfully transmitted, its originating node generates a new packet immediately. This may also correspond to nodes with long queues of packets to be transmitted.

Two cases are analyzed. First, the case in which the system is started under heavy load conditions is considered. This may not be a very practical situation, but it provides some significant insight into the system behavior. Secondly, the case where the system is started under light to medium load conditions, and then the load is increased until the system is saturated.

2.3.1. Heavy Load Startup

Let τ, T, and N respectively denote the retry time, packet length and number of nodes in a CASLAN operating under heavy load. Since the system starts heavily loaded, it is expected that all nodes would have packets transmitted to the hub at the same time. This is equivalent to synchronized packet transmission.

Theorem 2.5:

Let $T = k\tau + \Delta$, where $\Delta = T \bmod \tau$. Then the packet delay of CASLANs under heavy load conditions is exactly given by

$$D = NT + (N-1)\cdot(\tau-\Delta) + \tau$$

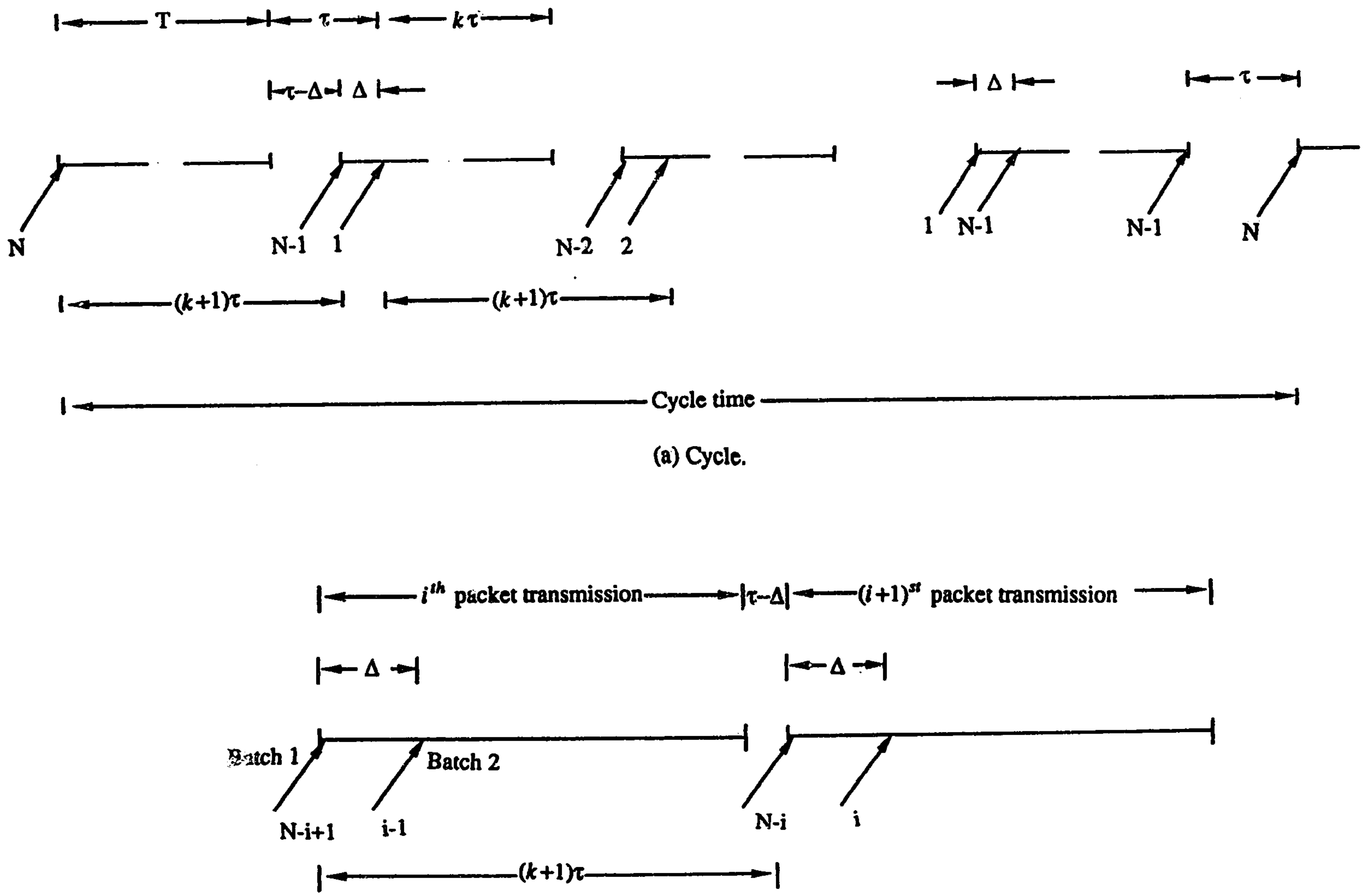
$$= (N-1)\cdot \left\lceil \frac{T}{\tau} \right\rceil \tau + T + \tau \qquad (2.19)$$

Proof:

Since the system is synchronized, all N nodes would have their packets arrive at the hub at almost the same time. Out of the N arriving packets, one of which would acquire the hub, the remaining $N-1$ packets would have their next interpacket arrival after time $(k+1)\cdot\tau$, see Figure 2.6.

One of the $N-1$ packets will acquire the hub. The node that has last released the hub would have a new packet arrive at the hub at time Δ after the hub acquisition. Next, it is shown that packet transmission proceeds in cycles and according to a first-come-first-served discipline. Also, during a cycle packets always pile into two batches, separated by time Δ from each other, one decreasing in size and other increasing, such that the sum of the sizes of the two batches is N. This process ends with the latter batch becoming of size N, ending the packet transmission cycle. This is proven by mathematical induction.

Assume that the cycle starts with all N packets arriving at the hub at the same time. Let the number of hub acquisitions, thus far, in the cycle be represented by i. For i=1, i.e. the first packet transmitted in the cycle, the size of batch #1 = N and the size of batch #2 = 0. As mentioned earlier, this leads to a state in which the size of batch #1 = $N-1$ and the size



(a) Cycle.

(b) The i^{th} and the $(i+1)^{st}$ packet transmissions.

Figure 2.6 : Hub cycle with $\tau \leq T$ and synchronized operation.

of batch #2 = 1, see Figure 2.6a.

Now assume that the i^{th} packet is being transmitted, i.e. the size of batch #1 $= N-i+1$ and the size of batch #2 $= i-1$. Since packet i acquired the hub, the size of batch #1 will decrease by 1. The remaining $N-i$ packets in the batch would have their next interpacket arrival after time $(k+1)\tau$. The packets of batch #2 would have their last intrapacket arrival after time $k\tau$. Since $T=k\tau+\Delta$, these packets would arrive exactly at the end of the current transmission and would not acquire the hub. In fact, the $(i+1)^{st}$ hub acquisition would be from one of the $N-i$ packets in batch #1, see Figure 2.6b.

During the $(i+1)^{st}$ packet transmission, batch #2 would be composed of all the packets of batch #2 that were present during the i^{th} packet transmission, as well as a newly generated packet by the originator of packet i (i.e. the node that has last released the hub). That is, batch #2 is of size $i-1+1=i$. It should be noted that the latter packet would arrive just after the arrival of $(i-1)$ older packets. This is due to the non-zero interarrival time. Therefore, the packets of batch #2 are ordered in a first-come basis.

For the N^{th} packet transmission, the size of batch #1 $=1$ and the size of batch #2 $=N-1$. That is, the sizes of batches #1 and 2 on the next transmission is 0 and N respectively. Since all N packets are in batch #2 one of them would acquire the hub. Thus, the hub will remain idle for time τ before the beginning of the next cycle (marked with all N packets arriving at the hub at the same time).

The cycle time is of length

$$D = NT + (N-1)\cdot(\tau-\Delta) + \tau$$

At steady state and during this cycle, packets are transmitted on a first-come-first-served basis. Since N packets are transmitted during the cycle, the cycle time represents the packet delay. This proves that the packet delay in a heavily loaded CASLAN is given by (2.19) above.

□

Note that the heavy load startup, or equivalently synchronized operation under heavy load results in a fixed and bounded packet delay. This is an important aspect in real-time applications, and will be dealt with later in this thesis. Lemmas 2.3 through 2.6 follow from theorem 2.5 above.

Lemma 2.3:

Let τ_f be a factor of T, and define $\varepsilon = 0^+$. Then, in a symmetric N-node CASLAN under heavy load conditions

$$D(\tau_f + \varepsilon) = NT + \tau_f \tag{2.20}$$

and

$$D(\tau_f - \varepsilon) = N\cdot(T + \tau_f) \tag{2.21}$$

Proof:

Recall that $\Delta = T \bmod \tau$. From the delay expression, equation (2.19), it can be easily seen that as Δ increases the delay decreases. For $\tau = \tau_f + \varepsilon$, $\Delta = \tau_f - \varepsilon$. Since $\varepsilon = 0$, then $D = NT + \tau_f$, which proves (2.20) above.

Now, for $\tau = \tau_f - \varepsilon$, $\Delta = \varepsilon$. That is,

$$D = NT + (N-1)\cdot\tau + \tau = N\cdot(T + \tau)$$

Since $\varepsilon = 0^+$, then $D = N\cdot(T + \tau_f)$, which proves (2.21).

□

Lemma 2.3 above shows that there exists a discontinuity in the delay function of CASLANs at heavy load and synchronized operation at retry time values that are factors of the packet length. Thus, it is more advantageous to operate CASLANs with a retry time that is slightly greater than a factor retry time value, viz., $(\tau_f + \varepsilon)$. For the remainder of this section, The term τ_f is used to denote $(\tau_f + \varepsilon)$. In the following lemma, the range of such retry time is shown.

Lemma 2.4:

Let τ_k be the retry time (a factor of T, such that $T = k\tau$) in a symmetric N-node CASLAN operating under heavy load conditions. The range of τ_k is given by

$$\left[\frac{\tau_k + (N-1)T}{N + (N-1)\dfrac{T}{\tau_k}}, \infty \right)$$

Proof:

From theorem 2.5, $D(\tau) = N\cdot T + (N-1)\cdot(\tau - \Delta) + \tau$. For τ within the range of τ_k, it follows that $D(\tau_k) \le D(\tau)$. This is clear for $\tau > \tau_k$. For $\tau < \tau_k$, $D(\tau_k) \le D(\tau)$ implies that:

$$N \cdot T + \tau_k \leq NT + \tau + (N-1)(\tau - \Delta)$$

or

$$\tau_k \leq N\tau + (N-1)\Delta \qquad (2.22)$$

Recall that

$$\Delta = T - k\tau$$

For $\tau > \tau_{k-1}$, where τ_{k-1} is the greatest factor of T less than τ_k, Δ is given by

$$\Delta = T\left(1 - \frac{\tau}{\tau_k}\right)$$

substituting in (2.22) above, it follows

$$\tau_k \leq N\tau + (N-1)T\left(1 - \frac{\tau}{\tau_k}\right) \qquad (2.23)$$

Rearranging the terms; (2.23) could be rewritten as

$$\tau \geq \frac{\tau_k + (N-1)T}{N + (N-1)\frac{T}{\tau_k}}$$

Thus the range of τ_k is given by

$$\left[\frac{\tau_k + (N-1)T}{N + (N-1)\frac{T}{\tau_k}}, \infty \right)$$

□

The above is an expression for the range of a factor retry time value under synchronized heavy load conditions. It is conjectured that the range is widest under heavy load and narrowest under very light load, $[\tau_k, \infty)$. For intermediate loads, the range would be some-where between these two extremes. It is interesting to note the effect of the number of nodes in CASLANs on the range of a factor retry time value, under heavy load conditions. For $N = 1$, the range is given by $[\tau_k, \infty)$, which should be expected, since the packet delay is given by $D(\tau) = T + \tau$. On the other end of the spectrum ($N \to \infty$), the range is given by

$$\left(\frac{T}{k+1}, \infty\right) = (\tau_{k-1}, \infty)$$

That is, the range of τ_k extends to, but does not surpass, τ_{k-1}.

Lemma 2.5:

Let τ_{f-1} and τ_f be two successive factors of T. Then, under heavy load conditions, the delay is linearly increasing with the retry time in the period $[\tau_{f-1}, \tau_f)$.

Proof:

From equation (2.19) above, $D(\tau)$ could written as

$$D(\tau) = T + (N + (N-1)\cdot\left\lfloor\frac{T}{\tau}\right\rfloor)\cdot\tau$$

For $\tau \in [\tau_{f-1}, \tau_f)$, $\left\lfloor\frac{T}{\tau}\right\rfloor$ is fixed. Thus in an N node CASLAN, the delay is of the form

$$D = a + b\tau$$

Hence the delay is linearly increasing with τ.

□

Lemma 2.6:

The utilization of an N-node CASLAN operating under heavy load conditions is given by:

$$U = \frac{N \cdot T}{N \cdot T + (N-1)(\tau - \Delta) + \tau} \tag{2.26}$$

Proof:

The proof follows directly from the fact that the cycle time of CASLANs (as was shown earlier) is given by:

$$C = N \cdot T + (N-1)(\tau - \Delta) + \tau$$

Since, N packets each of length T are successfully transmitted during a cycle, then equation (2.26) above is true.

□

From lemma 2.6 above, it can be seen that the utilization as a function of τ has local maxima when τ is a factor of T.

Corollary 2.1:

If τ is a factor of T, then the capacity of CASLANs is 1.

Proof:

For τ, a factor of T, the utilization is given by

$$U = \frac{NT}{NT + \tau}$$

For $N \to \infty$, $U_{max} \to 1$. That is, the capacity of CASLANs at such retry times is unity.

□

Next, some numerical results from the above analysis are shown. Figure 2.7 shows the delay versus the retry time for a 4, 10 and 20 node systems. Note that the delay is linearly increasing between every two factors (e.g. (120,160), (160,240) bits, etc.). Also note the sudden decrease in delay at retry times which are factors of τ. For instance, for the 20 node system at $\tau = 240$ bits. the delay dropped from 288 to 196.8, a 31.7% decrease. Note that the range of a factor retry time value increases with the number of nodes in the system. For example, at $\tau = 240$ bits, the range is given by $[168 \text{ bits}, \infty)$, $[162.9 \text{ bits}, \infty)$ and $[161.4 \text{ bits}, \infty)$ for $N = 4$, 10 and 20, respectively. It should be also noted that, since the delay is proportional to the packet length and the number of users, it is bounded if both of these parameters are finite.

Before leaving this section, the following observations are in order.

1- The order of packet transmission is kept because newly generated packets arrive at the hub just after ready packets. If this was not the case (due to any delays in the retry attempt of any ready packet) this order would not be kept and packet delays could become unbounded.

2- Recall from the proof of theorem 2.5, that during a cycle, two batches of packets exist. The second batch is the batch of newly generated packets ordered in a first-come basis. If any of the packets of this batch is transmitted[9] before the older packets of the first batch, the order of transmission will be disrupted and packet delays could again become unbounded. This again could be caused be any minor delay of

9. That is, have their last intrapacket arrival (which is just before the end of the current hub acquisition) become their next interpacket arrival, i.e., one of these packets would acquire the hub.

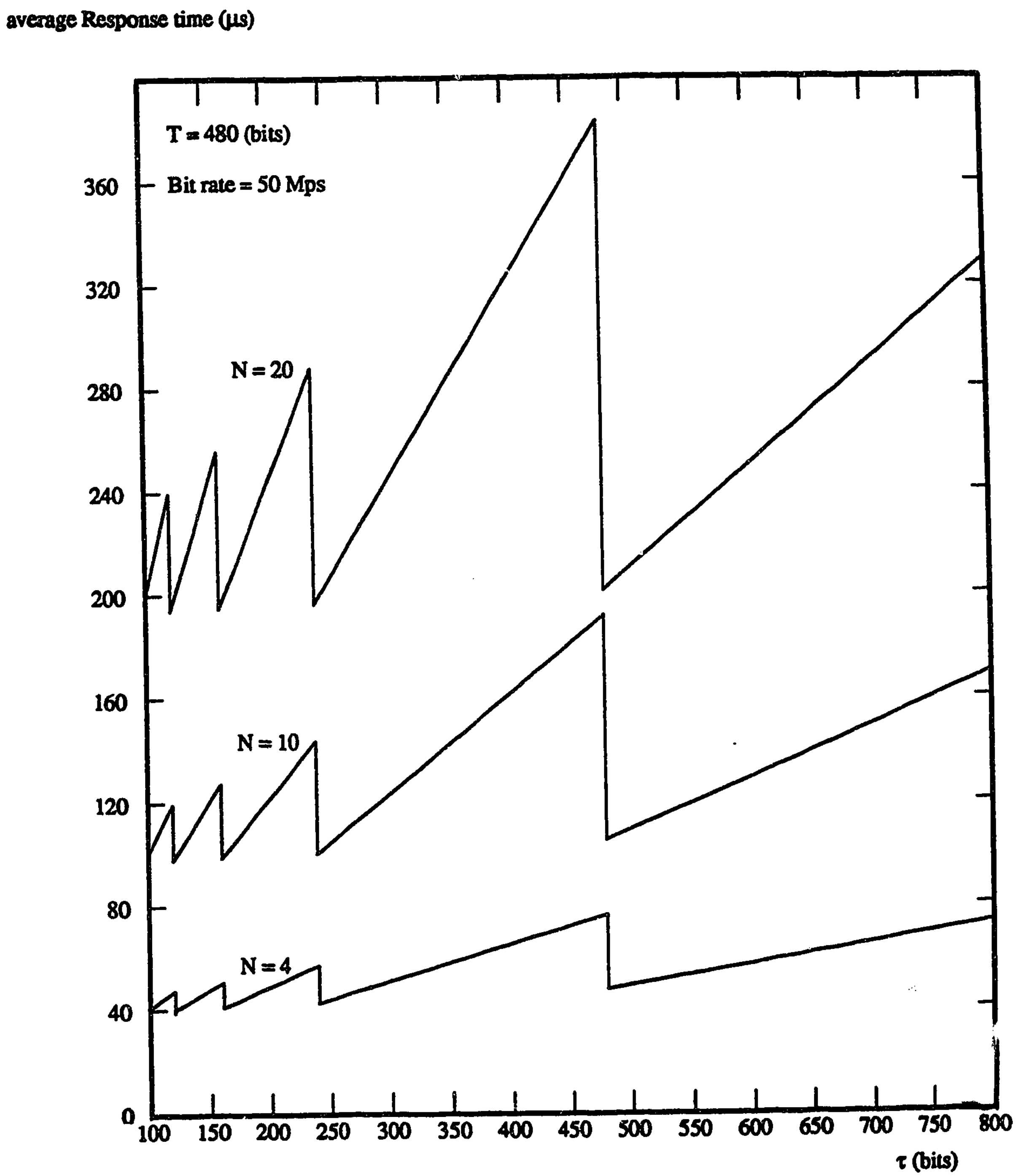


Figure 2.7: Response time at different values of τ (heavy load case)

any retry attempt. Such a deviation can be also caused by using different retry times for different nodes.

The implication of the above discussion is that timing is very critical in guaranteeing bounded delay in CASLANs. Hence, it is essential for the hub and the transmitting nodes to keep their timing constraints satisfied.

2.3.2. Light Load Startup

In this section, the behavior of CASLANs under heavy load conditions are analyzed with the assumption that the system was lightly loaded on startup and then the load is increased. This is the more practical case, since most systems do not startup very heavily loaded. The consequence of this mode of operation is that nodes will no longer be synchronized as in the case of section 4.1. First, the case where the retry time is a factor of the packet length is studied.

Theorem 2.6:

In a symmetric N-node CASLAN with retry time (τ_f) that is a factor of the packet length T, if the system becomes very heavily loaded, the packet delay is exactly given by

$$D = N \cdot T + \tau_f \tag{2.27}$$

Proof:

Let the system be moderately loaded, and let n out of N nodes be ready. If the system suddenly, or even gradually, becomes heavily loaded, then during the current packet transmission, all idle nodes would become ready, i.e., $N-1$ nodes would be ready.

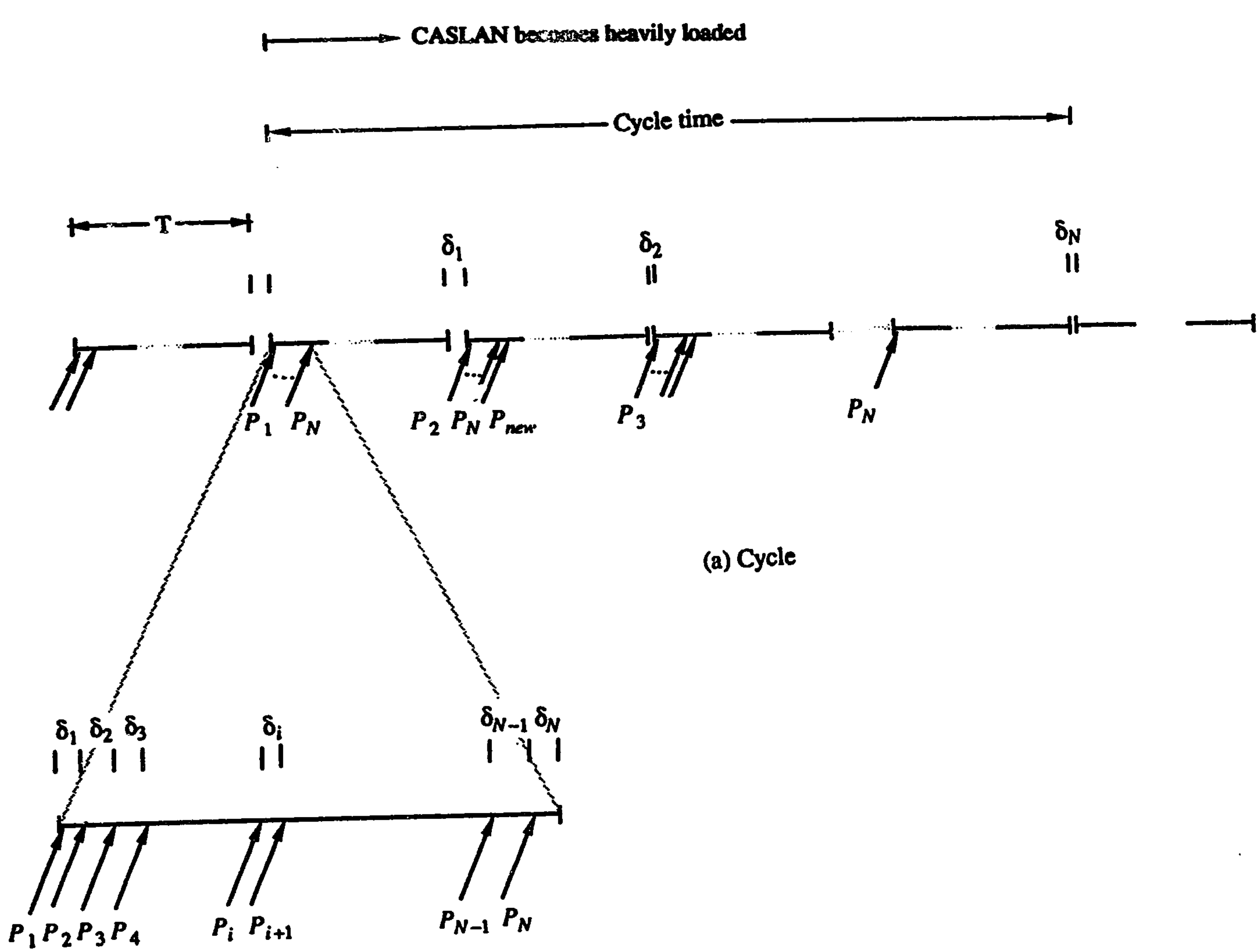
One of $N-1$ packets would acquire the hub next, the remaining $N-2$ packets would have their next interpacket arrival within time τ from the start of the hub acquisition. The node that last released the hub would have its new packet arrive after time τ from the hub release. Since there was a non-zero overhead period, the arrival of the new packet is still within time τ from the start of the current hub acquisition, see Figure (2.8a).

Since all nodes are ready, the overhead period is equal to δ, where δ is the arrival time of the first packet after the hub acquisition. From theorem 2.1, it is known that the inter-packet arrival time decreases by the overhead period (δ). This would imply that packets would be transmitted according to their order from the start of the hub acquisition. Since, new packets arrive at the hub after the ready ones, the packet transmission is on a first-come-first-served basis. In an N-node CASLAN, the cycle time in which all N packets are transmitted is given by

$$N \cdot T + \sum_{i=1}^{N} \delta_i \tag{2.28}$$

Where δ_i is the overhead after transmission of the i^{th} packet within the cycle.

Let the cycle start with the arrival of $N-1$ ready packets after the hub acquisition as shown in Figure (2.8b). It is not difficult to see that δ_1 is given by the difference in arrival time between the first and the second packet, δ_2 is given by the difference in arrival time between the second and the third packets, and in general δ_i, where $i = 1,2,...,N-1$, is the difference between the arrival of the i^{th} and the $(i+1)^{st}$ packets. δ_N is given by τ - the arrival time of the N^{th} packet. This means that



(a) Cycle

(b) Packet transmission order

Figure 2.8 : Hub Cycle (light load start-up), τ factor of T

$$\sum_{i=1}^{N} \delta_i = \tau \tag{2.29}$$

Thus, the cycle time is given by $N \cdot T + \tau$. Since N packets are transmitted during the cycle on a first-come first-served basis, the cycle time represents the packet delay. Hence, the packet delay is given by (2.27) above.

□

From the above theorem and theorem 2.5, it can be seen that, regardless of the system load on startup, if τ is a factor of T, then under heavy load conditions, the delay is given by $N \cdot T + \tau$.

It is interesting to note that, under heavy load conditions, the delay expressions in (2.27) above and (2.12) are the same. Under heavy load, where nodes are always ready, a packet from a station arrives at the hub τ units of time after the station releases the hub. For such a packet, and assuming a uniform distribution for retry attempts, the following is true

$$E[X_1] = T - \tau \quad \textit{(Approximate point of arrival of tagged packet)}$$

$$E[W] = \tau, \text{ and}$$

$$E[t] = \frac{\tau}{N-1} \quad \textit{(uniform distribution)}$$

Thus,

$$E[Z_2] = E[W] - E[t] = \tau - \frac{\tau}{N-1}$$

and,

$$E[D_Z] = T \cdot \frac{E[Z_2]}{E[t]} = (N-2) \cdot T$$

Substituting in (2.12), then

$$\bar{D} = T - \tau + \tau + (N-2)\,T + T + \tau$$
$$= NT + \tau$$

Note that, under heavy load conditions, D_Z is independent of τ. That is, the time spent attempting to acquire the hub is the same for all retry time values that are factors of the packet length, and is given by $((N-1)T)$.

Unfortunately, the result in theorem 2.6 above cannot be extended to the case where τ is not a factor of T. Recall that the proof of theorem 2.6, and for τ being a factor of T, depended on the fact that the interpacket arrival time for all packets decrease by the length of the overhead period, i.e., the order of arrival of packets is preserved. Since this is not the case for τ a-non-factor of T, the packet transmission is not on a first-come-first-served basis, and the packet delay could be unbounded. This was demonstrated in Lemma 2.2, and indeed, simulation results show that. Even though the average delay is reasonable, the delay can be unbounded. For real-time applications, however, where the probability of packets received before their deadlines is more important than maintaining a low average delay, bounding the delay is of extreme importance.

It might be worthwhile then to consider modifications to the original CASLANs protocol to bound the delay. One alternative is to synchronize the operation of CASLANs such that all packets are transmitted to the hub at the same time. Another alternative is to increase the retry time to the smallest possible factor of the packet length. Such modifications to the operation of CASLANs, along with others for variable packet length CASLANs, are described in [95] and [96].

2.3.3. CASLANs With Large Retry Time Values

In this section, the case in which the retry time is greater than the packet length is considered. Such a situation exists when the separation between nodes and the hub is large (i.e. CASLANs with a large radius), packets are very short, or both. It is shown that the delay could also be bounded in this case.

Theorem 2.7:

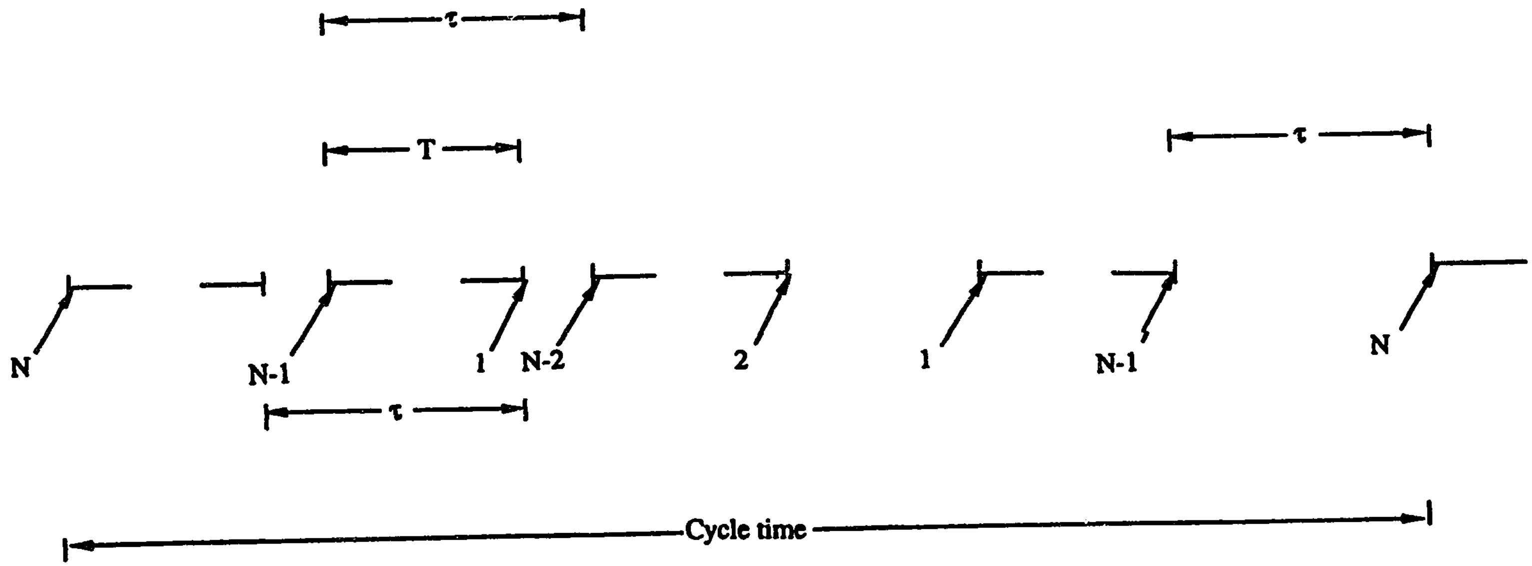
Let τ, T be the retry time and packet length in a symmetric N-node CASLAN. If $\tau > T$, then, under heavy load and synchronized CASLANs operation, the packet delay is exactly given by

$$D = T + N\tau \tag{2.30}$$

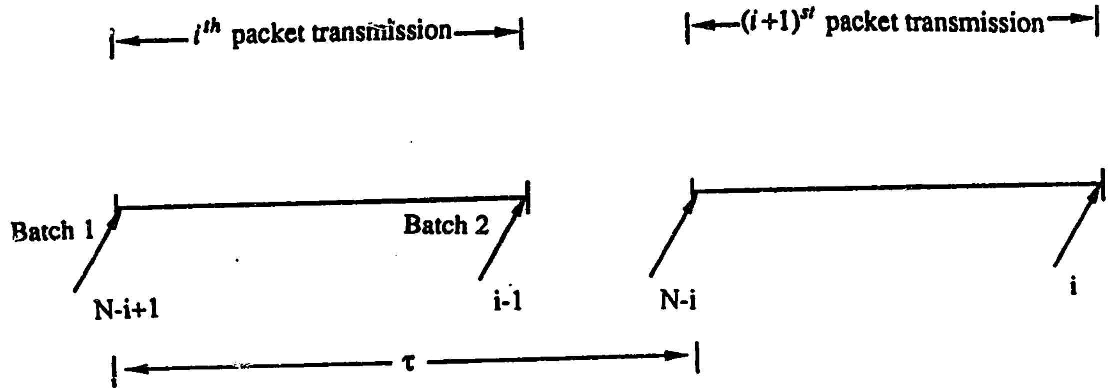
Proof:

Under synchronized operation, all N nodes would become ready and have packets arrive at the hub at the same time. One of these N packets acquires the hub, while the other $N-1$ packets would have their next interpacket arrival after time τ. One of the $N-1$ packets would acquire the hub (after time τ- T from the hub release). The node that has released the hub last would have a new packet arrive at the hub after time τ from the hub release, i.e., just before the end of the current transmission, see Figure 2.9.

As in the proof of theorem 2.5, mathematical induction is used to prove the cyclic



(a) Cycle.



(b) The i^{th} and the $(i+1)^{st}$ packet transmissions.

Figure 2.9: Hub cycle with $\tau \geq T$

nature of packet transmission. Again, the existence of two hub packet batches (during a cycle) is established, one decreasing in size and the other increasing, such that the sum of the sizes of these two batches is N. It is shown that, during a cycle packets are transmitted in a first-come-first-served basis.

Assume that the cycle starts with all N packets arriving at the hub at the same time. Let the number of hub acquisitions, thus far, in the cycle be represented by i. For i=1 (first hub acquisition), the size of batch #1 = N and the size of batch #2 = 0. As described above, this state leads to a state in which the size of batch #1 = $N-1$ and the size of batch #2 = 1. Thus, the condition holds for i=1.

Now for the i^{th} hub acquisition, let the size of batch #1 = $N-i+1$ and the size of batch #2 = $i-1$, see Figure 2.9b. In a fashion similar to that in the proof of theorem 2.5, it could be shown, that this leads to a state in which the size of batch #1 = $N-i$ and the size of batch #2 = i, where the packets of batch #2 are ordered in a first-come basis.

For the N^{th} packet transmission, the size of batch #1=1 and the size of batch #2= $N-1$. That is, the sizes of batch 1 and 2 on the next transmission is 0 and N respectively. Since all N packets are in batch 2 one of them would acquire the hub. Thus, the hub will remain idle for time τ before the beginning of the next cycle (marked with all N packets arriving at the hub at the same time).

The cycle consists of N transmissions separated by $(N-1)\cdot(\tau-T)$ periods, and followed by a time τ in which the hub is free. The cycle length is equal to

$$NT + (N-1)\cdot(\tau - T) + \tau = T + N\tau$$

Since all N packets are transmitted during this cycle in a first-come first-served basis, the cycle time represents the packet delay. This proves that the delay is as given by (2.30) above.

□

It should be noted that the delay linearly increases with the retry time and the number of users, and is bounded if these parameters are finite. Thus, by synchronizing the operation of CASLANs, bounded delay is achieved for all retry time values. Interestingly, and as should be expected, we note that if $T = \tau$, then equations (2.19) and (2.30) reduce to the same thing.

2.4. Summary

In this chapter an in-depth study on the effect of the retry time on the behavior of CASLANs has been conducted. Also, an exact analysis of CASLANs, under heavy load conditions, has been introduced.

Simulation results show that, in contrast to what was believed earlier, increasing the retry time does not always mean an increase in delay. In fact, the results show that increasing the retry time to a value which is a factor of the packet transmission time could result in a much better performance in terms of the average delay and throughput.

By analyzing the behavior of CASLANs in which the retry time is a factor of the packet length, the following have been shown:

(1) The delay at a retry time that is a factor of the packet length, is better than every higher retry time, and some lower retry time values subject to its range.

(2) A packet in a CASLAN with a retry time that is a factor of the packet length, is guaranteed transmission.

(3) Under heavy load conditions, a symmetric CASLAN with a retry time that is a factor of the packet length will have a bounded delay.

Heavy-load analysis of symmetric CASLANs with fixed-packet length show that

(1) Under synchronized operation, the delay is bounded and is exactly given by

$$D = \begin{cases} (N-1)\cdot \left\lceil \frac{T}{\tau} \right\rceil \tau + T + \tau & \tau \le T \\ T + N\tau & \tau > T \end{cases}$$

(2) For retry time values that are factors of the packet length the packet delay is given by $(NT+\tau)$ and behaves as a local minima.

(3) The capacity of symmetric CASLANs, with a retry time that is a factor of the packet length, approaches unity.

Chapter 3

Performance Modeling of CASLANs

The purpose of this chapter is to introduce a new performance model for CASLANs. The difference between this model and earlier models for CASLANs is that this model is capable of capturing the phenomenon described in chapter 2, namely, the dependence between message transmission and retry times. Moreover, the model proves to be more accurate than all previously devised models reviewed in the following section.

3.1. Previous Work

Several attempts have been made to study the performance of CASLANs. The attempts fall into three categories: mathematical analysis [39-45], simulation [46-47], and experimental measurements [48].

In [40] and [41] a geometric process was used to model the access protocol of CASLANs. The blocking probability was taken as the system utilization. If $\bar{T}_i$ and τ_i are the average packet length and the retry time of node i, respectively, then the average delay of node i, $\bar{D}_i$, is given by:

$$\bar{D}_i = \frac{\tau_i}{1-\rho} + \bar{T}_i$$

Where $0 \le \rho < 1$, is the channel utilization.

The model in [39] is very similar, except that the author used the packet transmission time instead of the retry time as the minimum time to acquire the channel. The average delay expression in this case is given by:

$$\bar{D} = \frac{T + \tau}{1 - \rho}$$

Suda and Goto [42] modified the above approach by estimating the average number of retry attempts, $\bar{m}$. In their approach, they assumed a slotted system in which the slot length is equal to the packet length. The retry time τ was taken as R slots. Idle users were modeled to arrive via a geometric process and ready users were assumed to access one of R+1 slots with a uniform probability, $\frac{1}{R+1}$. With this model the average delay is given by

$$\bar{D} = \bar{m}\ (R+1) + R + 1$$

A different approach was adopted by Kamal and Hamacher [44] and later improved by Kamal [45]. In [44] and [45], CASLANs are modeled as polling systems where the hub chooses a node at random from the idle or ready nodes. Two service times are separated by an overhead period whose length is dependent on the number of ready nodes and the offered load. The system was modeled using an embedded Markovian chain, in which the states correspond to the number of ready users in the system. By making use of the regenerative nature of Markovian chains, the throughput, X, is obtained. The average delay, $\bar{D}$, is then given by

$$\bar{D} = \frac{N}{X} - \frac{1}{\lambda}$$

The underlying assumptions behind the model in [44] were:

(1) Packets from ready nodes are given priority over packets from idle ones. That is,

given that at least one node is ready at the end of the current hub acquisition, the hub will next be acquired by one of these ready nodes. This assumption is not necessarily true, since, and as was noted in [44], a packet arrival from an idle node may acquire the hub, despite the presence of other ready nodes.

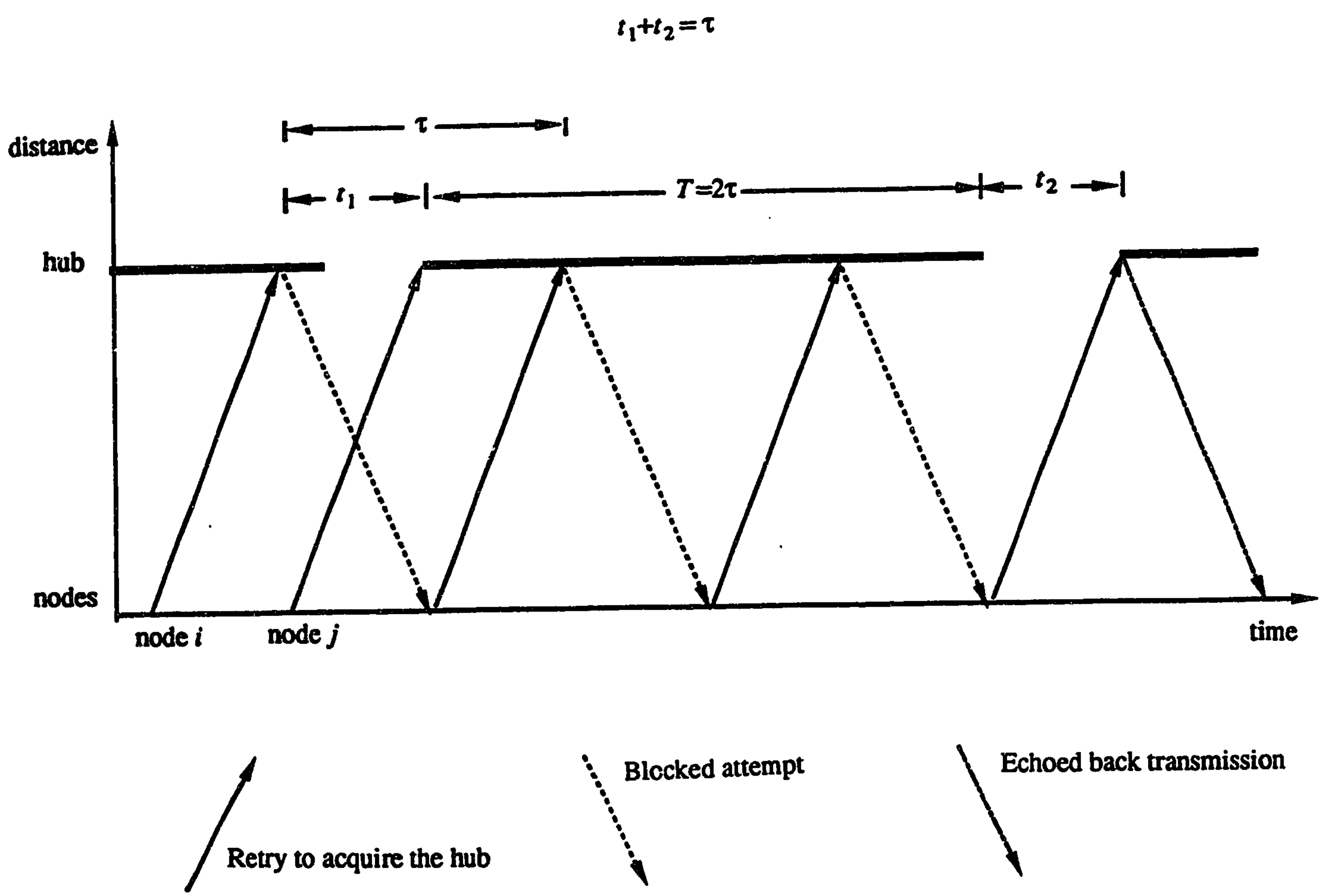


Figure 3.1: Dependence between retry instants and the packet length for T a multiple of τ

(2) The retry instants are independent of each other and of the packet length. In Figure 3.1, the retry instants of a packet transmission, in a CASLAN in which the retry time is a factor of the packet length, are shown. Note that $t_1+t_2=\tau$, as $t_1+t_2+T=3\tau$.

The model in [45] relaxed assumption (1) above, by allowing packets from idle nodes to contend for the hub in the presence of ready nodes. The model was shown to be more accurate than that in [44] and, in fact, than all previous models [39-43].

In [46], a simulation model for CASLANs was introduced and discussed. However, no results were reported. Another simulation study was conducted in [47]. This was a comparative study between a 100 Mbps Hubnet[10] and FDDI[11] [49]. Different values of the number of nodes, packet lengths, packet length distributions, network dimensions and offered loads were tested. The authors showed that Hubnet always outperforms FDDI under light to high offered load conditions (the heavy load case was not considered).

An experimental study of CASLANs was conducted in [48]. Actual performance measurements of Hubnet were observed. The effect of the offered load and the distribution of packet length were extensively studied. The study lacked the following:

(1) Results of a very heavily loaded (congested) network, (the maximum system utilization observed was 0.875).

10. Hubnet [37] is a CASLAN that was developed at the University of Toronto in Canada, in cooperation with CANSTAR corporation. Hubnet uses glass fiber as its transmission medium and is operated at a data rate of 50 Mbps, that was recently upgraded to 100 Mbps.

11. Fiber Distributed Data Interface.

(2) A study of the effect of the retry time on the performance of CASLANs.

In chapter 2, the behavior of fixed-packet-length CASLANs was analyzed. The above two points, in particular, were extensively studied. It was shown that the average delay in CASLANs does not always increase with increasing retry time. For retry time values that are factors of the packet length, the average delay is less than that at some lower retry time values that fall within the retry time's range. This range was shown, through simulation, to be mainly dependent on the offered load and, indeed, increases with the load.

Figure 3.2 shows the average delay versus the retry time of CASLANs, with N=20 nodes and T=480 bits, at different offered loads. The results are compared to those in [45]. From the results in the figure, it can be observed that the phenomenon above is not captured by any of the previous performance models [39-45]. This, as will be shown, is mainly due to assumption 2 above.

In this part of the thesis, a new CASLANs performance model is devised. The model should be more accurate than previous models and capable of capturing the phenomenon discovered in the previous chapter.

3.2 The Model

The model presented in this section is a 3-dimensional Markovian chain, and is based on following a tagged user in an exact analytical approach, while the treatment of the rest of the users is through an approach that involves some simplifying approximations.

To be able to see why such approximations are necessary, one has to study the feasibil-

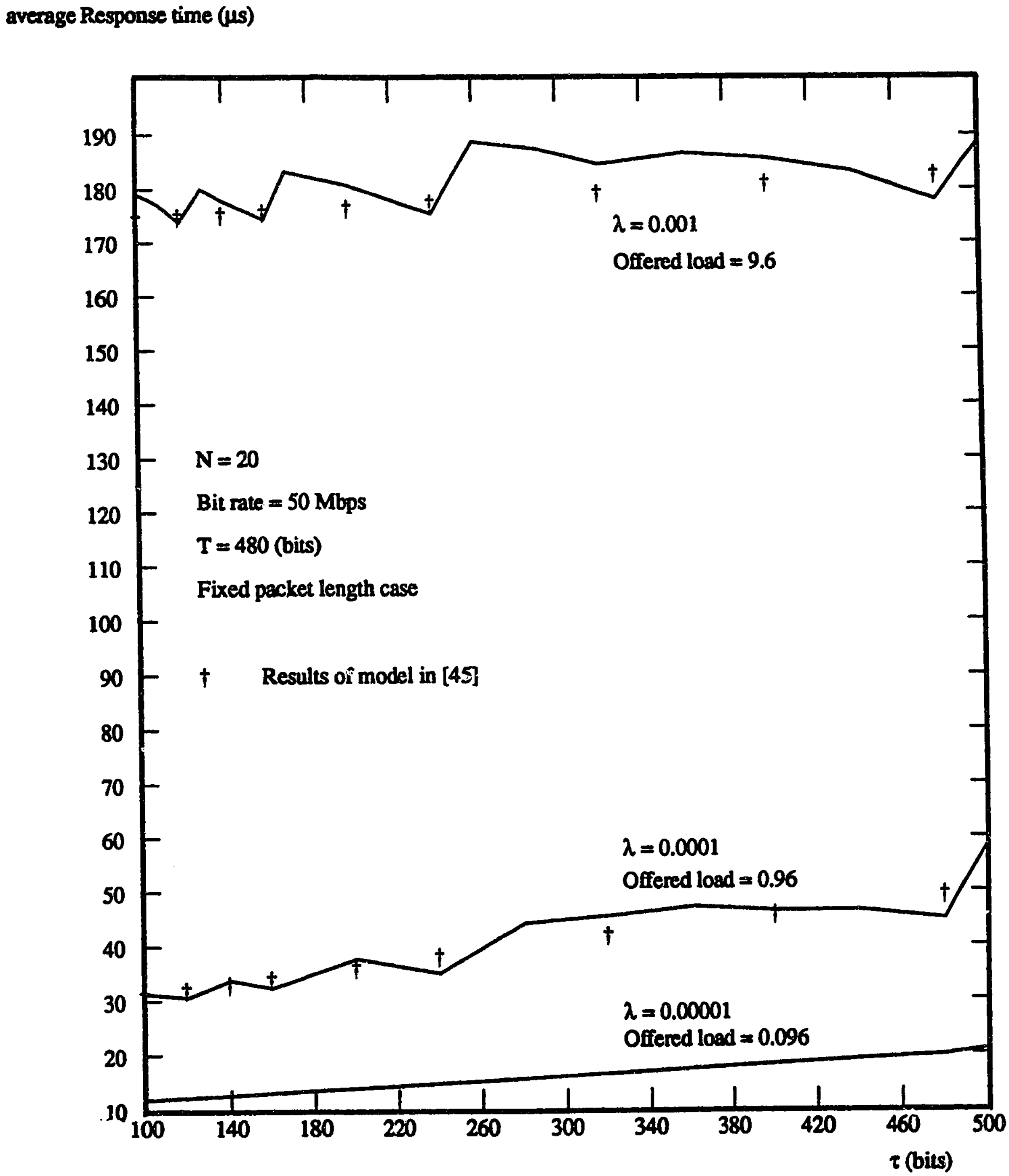


Figure 3.2: Simulation results compared to the results of the model in [45]

ity of an exact CASLANs analytical model. For an exact CASLANs analysis, a model which keeps track of interpacket arrival times for all nodes must be devised. This means that using a Markovian chain approach, a dimension must be added to the state space for every node in the system. For any reasonable number of nodes, such a model is computationally expensive, let alone its extreme complexity.

Having demonstrated the complexity of an exact analytical model, we next outline our assumptions.

Assumptions used:

(1) Except for the tagged user, the retry instants for packets of all nodes are independent of each other and of the packet length, and are identically distributed according to a general, unknown distribution.

(2) All nodes are equipped with single packet occupancy buffers. That is, no new packets are generated at a node unless the packet currently in the buffer has been successfully transmitted.

(3) Identical and independent packet generation processes at all nodes. The packet generation time is exponentially distributed with rate λ.

(4) Fixed and equal packet length (T) for all nodes.

(5) The round trip propagation delay between any node and the hub is the same for all nodes, and is equal to the retry time (τ).

(6) $T \geq \tau$

The model in this chapter is based on a discrete time approach. The retry time is divided into a number of slots, g, which is referred to as the grid size. Of course, the larger the grid size, or g, the closer the operation to that of an unslotted system, and the more accurate the model. This, however, is done at the expense of increasing the state space. An embedded Markovian chain model is used with the embedding points being at the start of hub acquisitions. This model evolved from a continuous time model, which is described in Appendix 3.A. The latter, however, involves dealing with a continuous time Markovian chain for which obtaining steady state measures is much harder and requires the numerical solution of integral equations.

The performance measures are

(1) the system utilization defined as the percentage of the time the hub is utilized, and

(2) the average packet delay, which is measured from the time the packet is generated until the time the packet arrives at the destination.

Let t, Z, X and Y be as defined in section 2.2. In this model, the overhead period, t, as well as the values of Z, X and Y are given in terms of slots. Given the grid size, g, the slot size in bits, ζ, is given by

$$\zeta = \frac{\tau\,(bits)}{g}$$

The grid size would then represent the retry time in slots. The packet length, T, is also given in slots, and is equal to $K\tau + \Delta$, where,

$$K = \left\lfloor \frac{T}{\tau} \right\rfloor$$

and

$$\Delta = T \ mod \ \tau.$$

An idle user may generate a packet in a slot with probability σ, where

$$\sigma = 1 - e^{-\lambda\zeta},$$

where λ is the packet arrival rate per bit time. Thus, instead of the exponential distribution, and because of the discrete nature of the model, a geometric distribution for packet arrival results. Packets arriving at the hub during a slot are delayed to the beginning of the next slot. That is, packets contending to acquire the hub are those arriving during the last slot of the overhead period.

In this model, the system is represented as a polling one. Since this model is a discrete one, the hub, which is the server in the system, randomly selects one of the packets that arrive in the first empty slot after it has been released. If no packets arrive in this slot, the hub inspects the next slot. This inspection and selection operation is repeated until the hub is finally acquired. This means that the overhead period is at least one slot. To facilitate the analysis, users who may acquire the hub are divided into four groups (see Figure 3.3):

- Idle users. These are the users with no packets to transmit at the end of the hub acquisition. They may then generate a packet in an idle slot before the next hub acquisition. If they do, they may only acquire the hub at the beginning of the following slot. Idle users are modeled exactly.

- First-time ready users. Those are the users which became ready during the last hub acquisition, and are modeled exactly.
- Carry-over ready users. Those are the ready users that were ready before the start of the last hub acquisition. This is the only class of users that is modeled approximately.
- The tagged user, which may or may not be ready and is modeled exactly.

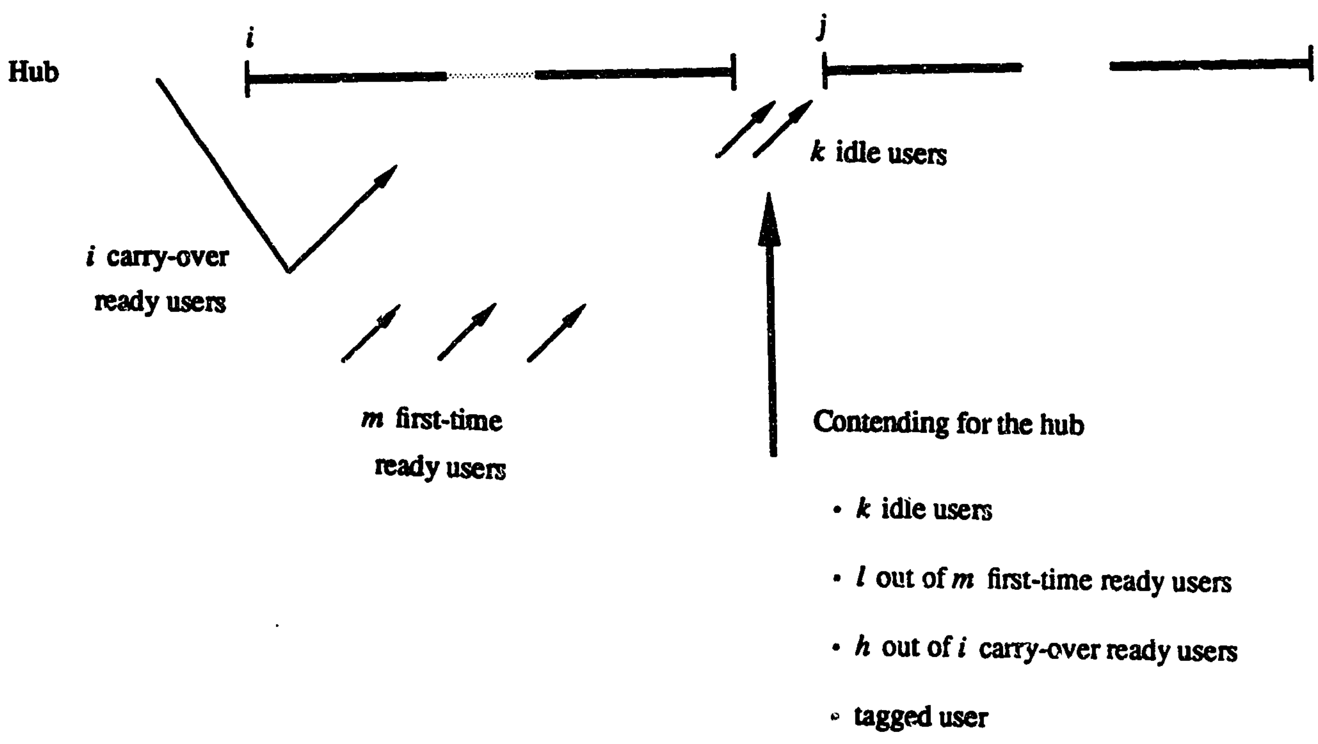


Figure 3.3 : Illustration of the different classes of users contending for the hub

Idle users may generate packets according to a geometric process with parameter σ, and are modeled exactly. Carry-over ready users are initially[12] modeled to obey a uniform distribution, where the arrival instant at the hub of the start of a packet transmission is chosen randomly from the period $(0,\tau]$, following the end of a transmission. First-time ready users are modeled exactly by making use of the retry nature of CASLANs, see Appendix 3.B.

Note that the last user to release the hub must wait for a time τ before it may generate a new packet ($\frac{\tau}{2}$ from the hub release to the node, and $\frac{\tau}{2}$ before the hub acquisition from the node to the hub). If the last user that released the hub becomes ready during the following hub acquisition, it joins the carry-over ready users. That is, it is not treated as a new[13] arrival. Finally, the tagged user is modeled exactly by using the Y[14] values as a reference of interpacket arrival times. (Refer to Figure 2.3.)

A three-dimensional embedded Markovian chain is used to represent the polling system, where the embedding points are at the start of hub acquisitions. Note that this is a fundamental and necessary difference from all the other models that embed the chain at the end of the hub acquisition. The dimensions of the chain are as follows:

1. The first dimension is the state in which the tagged user is in. We consider three different states for the tagged user; namely: $\bar{R}$ (not ready), A (Acquiring the hub)

12. In section 3.3, a better approximation for carry-over ready users is presented using the steady state behavior of the tagged user. However, an initial distribution must be used to represent these users. As we shall see, this initial distribution has little effect on the final results of the model.

13. In fact, if it is desired to treat such a user as a new arrival, information regarding the length of the overhead period during the last transition would be required. This information cannot be incorporated in our Markovian chain model.

14. The Z values could have been used alternatively.

and R (ready).

2. The second dimension is the number of carry-over ready nodes.

3. The third dimension is the remainder of the hub acquisition time at the instant of the last arrival of the tagged users packet (Y), if the tagged user is ready. If the tagged user is not ready, we arbitrarily set Y to zero.

In such a representation, $P(new, j, y_l \mid old, i, y_{l-1})$ is the transition probability from state (old, i, y_{l-1}) to state (new, j, y_l), where

old (*new*)	is the old (or new) state of the tagged user.
i (j)	is the number of carry-over ready users before (and after) the transition.
y_{l-1} (y_l)	is the value of Y of the tagged user's packet before (and after) the transition. In the sequel, y_{l-1} and y_l are referred to as y_{bef} and y_{aft} corresponding to the values of Y before and after the transition, respectively.

Note that the value of Y is relevant only when the tagged user is ready, and set to zero otherwise. Also note that, since the first state variable assumes only one of three states $(\bar{R}, A, R)$, it does not change the order of the state space.

Before introducing the transition probabilities, we make the following definitions.

- $P_x(u) = 1-(1-\sigma)^u$

This is the probability that a packet is generated from an idle node during time u.

- $$P_{tx}(n,M,T) = \begin{cases} \binom{M}{n} P_x(T)^n \, (1-P_x(T))^{M-n} & 0 \le n \le M \\ 0 & \text{otherwise} \end{cases}$$

This is the probability of n packet generations from M idle nodes during the transmission time, T.

- $$P_L(t) = \begin{cases} P_x(t+T-\tau) & 0 < t \le \tau \\ (1-\sigma)^{t-\tau} P_x(T) & t > \tau \end{cases}$$

This is the probability that the last node to release the hub will transmit during the following hub acquisition that occurs after an overhead period of length t.

- $P_{\bar{L}}(t) = 1 - P_L(t)$

- $$P_{idle}(k,M,t) = \begin{cases} \binom{M}{k} \sigma^k \, (1-\sigma)^{Mt-k} & 0 \le k \le M \\ 0 & \text{otherwise} \end{cases}$$

This is the probability that k packets out of M idle users arrive at the hub exactly in slot t.

$$\bullet\; P_{ready}(h, M, t) = \begin{cases} \binom{M}{h} \dfrac{(\tau - t)^{M-h}}{\tau^M} & 0 \le h \le M \;\&\; t < \tau \\ \dfrac{1}{\tau^M} & h = M \;\&\; t = \tau \\ 0 & otherwise \end{cases}$$

This is the probability that h packets out of M carry-over ready users arrive at the hub exactly in slot t. This expression is based on using a uniform distribution to represent the arrival point at the hub of carry-over ready users. In section 3.3, another expression for $P_{ready}(h, M, t)$ is derived, using a more elaborate approximation for carry-over ready users.

$$\bullet\; P_{new}(R, M, t) = \begin{cases} \binom{M}{l} P_a(t)^l \, (1 - P_b(t))^{M-l} & 0 \le l \le M \;\&\; t < \tau \\ P_a(t)^M & l = M \;\&\; t = \tau \\ 0 & otherwise \end{cases}$$

This is the probability that l packets out of M new (first-time ready users) arrive at the hub at exactly time t. $P_a(t)$ is the probability that a new user's packet arrives at the hub exactly in slot t after the hub release. An expression for $P_a(t)$ is given in Appendix 3.B. $P_b(t)$ is the probability that a packet from a new user arrives at the hub during time t, and is given by

$$P_b(t) = \sum_{i=1}^{t} P_a(i)$$

The above probabilities are all functions of t, the overhead period. Since t is dependent on the number of ready users in the system, as well as the interarrival times of the tagged user, the joint transition probabilities $P^{(t)}(new, j, y_{aft} \mid old, i, y_{bef})$ must be defined first, then summed over all values of t. Depending on the state of the tagged user, 9 different cases exist. Table 3.1 gives the number of first-time ready (new) users involved in a transition.

	Hub acquired by one of			
	k out of N-i-m idle users	h out of i carry-over ready users	l out of m new arrivals	tagged user
($\bar{R}$,i) last will not become ready	j-i-k+1	j-i-k+1	j-k-i+1	j-k-i
($\bar{R}$,i); last will become ready	j-k-i	j-k-i	j-k-i	j-k-i-1
(A,i)	j-i-k+1	j-i-k+1	j-k-i+1	0
conditions	$k \geq 1$	$i \geq 1$	$m \geq 1$	$? \rightarrow A$ transition

An $R \rightarrow ?$ is similar to an $\bar{R} \rightarrow ?$ transition.

Table 3.1 : Number of new arrivals (m) during the current transmission (given i carry-over ready users and k packets from idle users contend for the hub)

I. Tagged user is initially not ready:

<u>I.i Case $\bar{R} \rightarrow \bar{R}$:</u>

The tagged user is not ready and will not become ready during the transition. Two ranges of t are distinguished.

- $1 \le t \le \tau$

$$P^{(t)}(\bar{R},j,y_{aft} \mid \bar{R},i,y_{bef}) = (1-P_x(t+T)) \cdot \sum_{k=0}^{j-i+1} \sum_{l=0}^{j-k-i+1} \sum_{h=0}^{i}$$

$$\Bigg\{ P_{\bar{L}}(t) \cdot P_{tx}(N-i-1, j-k-i+1, T) \cdot P_{idle}(k, N-j+k-2, t) \cdot P_{ready}(h, i, t) \cdot P_{new}(l, j-k-i+1, t)$$

$$+P_L(t) \cdot P_{tx}(N-i-1, j-k-i, T) \cdot P_{idle}(k, N-j+k-1, t) \cdot P_{ready}(h, i, t) \cdot P_{new}(l, j-k-i, t) \Bigg\} \qquad (3.1)$$

The term outside the summation in equation (3.1) above is the probability that the tagged user does not generate a packet during the transition. The term inside the curly bracket is the probability that k idle, h carry-over ready and l new users have their packets arrive at the hub exactly in slot t after the hub release. Note that only one of these users would acquire the hub. The rest, if any, will become carry-over ready users at the next embedding point.

- $t > \tau$

For $t > \tau$, there should be no carry-over ready users ($i=0$), nor should there be any first-time ready users[15]. Also, the last user to release the hub is allowed to contend for it.

15. Because of retry nature of CASLANs, and as mentioned earlier, packets from ready users cannot arrive at the hub at a time later than the retry time, τ.

$$P^{(t)}(\bar{R},j,y_{aft}\mid\bar{R},0,y_{bef})=(1-P_x(t+T))\cdot P_{tx}(N-1,0,T)\cdot$$

$$\left[\sigma(1-\sigma)^{t-\tau-1}\cdot P_{idle}(j,N-1,t)+P_L(t)\cdot P_{idle}(j,N-1,t)+P_{\bar{L}}\cdot P_{idle}(j+1,N-1,t)\right] \tag{3.2}$$

The two terms outside the square brackets are the probabilities that the tagged user does not become ready and the probability of no new arrivals, respectively. The term inside the square brackets is the probability that $j+1$ users contend for the hub exactly t slots after the hub release. This probability takes into account whether the last user to release the hub is one of these $j+1$ users (the first term), or not (the last two terms).

<u>I.ii Case $\bar{R}\rightarrow A$</u>

The tagged user acquires the hub during the transition. The transition probability is given by

$$P^{(t)}(A,j,y_{aft}\mid\bar{R},i,y_{bef})=\sigma(1-\sigma)^{t-1}\cdot\sum_{k=0}^{j-i}\sum_{l=0}^{j-k-i}\sum_{h=0}^{i}\frac{1}{k+l+h+1}$$

$$\Big\{P_{\bar{L}}(t)\cdot P_{tx}(N-i-1,j-k-i,T)\cdot P_{idle}(k,N-j+k-1,t)\cdot P_{ready}(h,i,t)\cdot P_{new}(l,j-k-i,t)$$

$$+P_L(t)\cdot P_{tx}(N-i-1,j-k-1,T)\cdot P_{idle}(k,N-j+k,t)\cdot P_{ready}(h,i,t)\cdot P_{new}(l,j-k-i-1,t)\Big\} \tag{3.3}$$

The term outside the summation in equation (3.3) above is the probability that the tagged user packet arrives exactly in slot t. The summation is over all possible combinations of having k idle, h carry-over ready and l new users schedule their packet arrivals at the hub, exactly t slots after the hub release. Again, the probability accounts for both cases in which the user that last released the hub is involved or not.

<u>I.iii Case $\bar{R}\to R$</u>

The tagged user becomes ready. Let Z be the first interarrival time of the packet from the tagged user. Now $X=T-Z$, see Figure 3.4. Two distinct ranges of X can be identified.

a) $1\le X\le T$:

This is the normal case where the tagged user's packet arrival finds a busy hub, see Figure 3.4a. The transition probability is given by

$$P^{(X,t)}(R,j,y_{aft}\mid\bar{R},i,y_{bef})=\sigma(1-\sigma)^{t+T-X-1}\cdot\sum_{k=0}^{j-i+1}\sum_{l=0}^{j-k-i+1}\sum_{h=0}^{i}$$

$$\Big\{P_{\bar{L}}(t)\cdot P_{tx}(N-i-1,j-k-i+1,T)\cdot P_{idle}(k,N-j+k-2,t)\cdot P_{ready}(h,i,t)\cdot P_{new}(l,j-k-i+1,t)$$

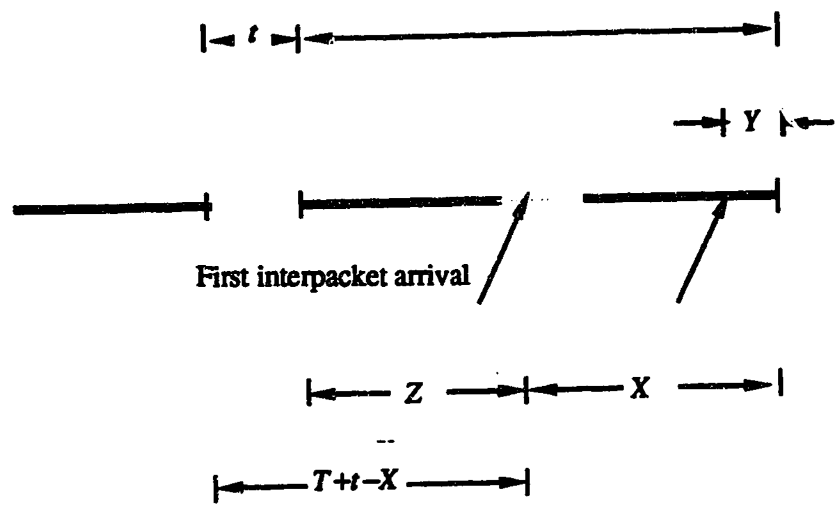
$$+P_L(t)\cdot P_{tx}(N-i-1,j-k-i,T)\cdot P_{idle}(k,N-j+k-1,t)\cdot P_{ready}(h,i,t)\cdot P_{new}(l,j-k-i,t)\Big\} \quad (3.5)$$

The term outside the summation is the probability that the tagged user becomes ready at exactly slot $t+T-X$. The terms inside the curly brackets are the same as those in (3.1).

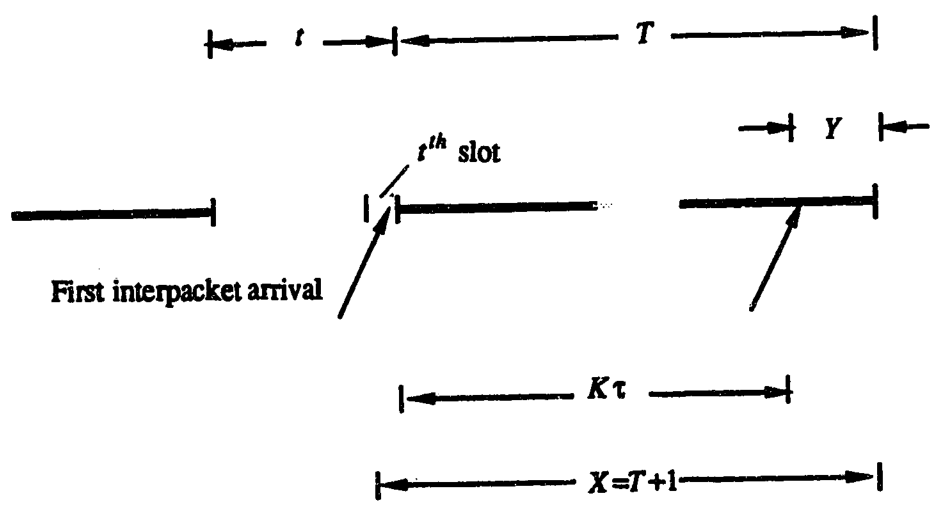
b) Case $X=T+1$

This case is different from that in a) because the tagged user will contend for the hub, see Figure 3.4b. The transition probability is given by

$$P^{(X,t)}(R,j,y_{aft}\mid\bar{R},i,y_{bef})=\sigma(1-\sigma)^{t-1}\cdot\sum_{k=0}^{j-i+1}\sum_{l=0}^{j-k+1}\sum_{h=0}^{i}\frac{k+l+h}{k+l+h+1}\cdot$$



(a) $X \leq T$



(b) $X = T + 1$

Figure 3.4 : Arrival of the tagged user's packet in an $\bar{R} \rightarrow R$ transition

$$\Big\{P_{\bar{L}}(t)\cdot P_{tx}(N-i-1,j-k-i+1,T)\cdot P_{idle}(k,N-j+k-2)\cdot P_{ready}(h,i,t)\cdot P_{new}(l,j-k-i+1,t)$$

$$+P_L(t)\cdot P_{tx}(N-i-1,j-k-i,T)\cdot P_{idle}(k,N-j+k-1,t)\cdot P_{ready}(h,i,t)\cdot P_{new}(l,j-k-i,t)\Big\} \qquad (3.6)$$

Equation (3.6) differs from equation (3.5) in one term only, which is the quotient outside the curly brackets, since this accounts for the total number of users contending for the hub, which, in this case, includes the tagged user.

Now, recalling that $X = n\tau + y_{aft}$ and summing over all values of X, then

$$P^{(t)}(R,j,y_{aft} \mid \bar{R},y_{bef}) = \begin{cases} \sum_{n=0}^{K} P^{(n\tau+y_{aft},t)}(R,j,y_{bef} \mid \bar{R},i,y_{bef}) & 0 \le y_{aft} < \Delta \\ \sum_{n=0}^{K-1} P^{(n\tau+y_{aft},t)}(R,j,y_{aft} \mid \bar{R},i,y_{bef}) & \Delta \le y_{aft} < \tau \end{cases} \qquad (3.7)$$

II. Tagged user is initially acquiring the hub

<u>II.i Case $A \rightarrow \bar{R}$:</u>

$$P^{(t)}(\bar{R},j,y_{aft} \mid A,i,y_{bef}) = (1-P_x(t+T-\tau))\cdot \sum_{k=0}^{J-i+1} \sum_{l=0}^{j-k-i+1} \sum_{h=0}^{i}$$

$$P_{tx}(N-i,j-k-i+1,T)\cdot P_{idle}(k,N-j+k-1)\cdot P_{ready}(h,i,t)\cdot P_{new}(l,j-k-i+1,t) \qquad (3.8)$$

The term outside the summation in equation (3.8) represents the probability of no packet arrival from the tagged user. The term inside the summation is the probability that k, h and l packets from the idle, carry-over ready and first-time ready users, respectively, arrive at exactly slot t.

<u>II.ii Case $A \rightarrow A$</u>

For the tagged user to reacquire the hub on the very next hub acquisition, t must be greater than τ. Thus, there should be no ready users at the first embedding point. The transition probability is given by

$$P^{(t)}(A,j,y_{aft} \mid A,i,y_{bef}) = \sigma(1-\sigma)^{t-\tau-1} \cdot P_{idle}(j,N,t) \cdot P_{tx}(N,0,T) \cdot \frac{1}{1+j} \qquad (3.9)$$

In (3.9) above, the first term is the probability of a packet arrival from the tagged user at exactly slot t, the second term is the probability of the arrival of j packets out of N idle users, while the third is the probability of no new arrivals during the tagged user first transmission. The last term is the probability that the hub randomly chooses the packet from the tagged user over the other j packets that arrived in slot t.

II.iii Case $A \rightarrow R$

This case is similar to the $\bar{R} \rightarrow R$ case, except for two differences. First, the tagged user must wait for a time equal to τ before being able to generate its next packet. This is because the tagged user is the last to release the hub. Second, two cases must be distinguished. Namely, the cases in which $t \leq \tau$ and $t > \tau$.

- $1 \leq t \leq \tau$

Referring to Figure 3.5, it is not hard to see that $1 \leq X \leq T+t-\tau$. The transition probability is given by

$$P^{(X,t)}(R,j,y_{aft} \mid A,i,y_{bef}) = \sigma(1-\sigma)^{t+T-\tau-X} \cdot \sum_{k=0}^{J-i+1} \sum_{l=0}^{j-k-i+1} \sum_{h=0}^{i}$$

$$P_{tx}(N-i,j-k-i+1,T)\cdot P_{idle}(k,N-j+k-1)\cdot P_{ready}(h,i,t)\cdot P_{new}(l,j-k-i+1,t) \qquad (3.10)$$

- $t>\tau$

Here again a distinction between the case where $1\le X\le T$ and the case where $X=T+1$ is made. The latter represents the case where the tagged user contends for the hub. Note that because of the limits on X, this case was non-existent with $t\le\tau$.

a) $1\le X\le T$

The transition probability is exactly same as (3.10) above except for the limits on X.

b) $X=T+1$

The tagged user contends for the hub. The transition probability is given by

$$P^{(X,t)}(R,j,y_{aft}\mid A,0,y_{bef}) = \sigma(1-\sigma)^{t-\tau-1}\cdot\sum_{k=0}^{j-i+1}\sum_{l=0}^{j-k-i+1}\sum_{h=0}^{i}\frac{k+l+h}{1+k+l+h}\cdot$$

$$P_{tx}(N-i,j-k-i+1,T)\cdot P_{idle}(k,N-j+k-1)\cdot P_{ready}(h,i,t)\cdot P_{new}(l,j-k-i+1,t) \qquad (3.11)$$

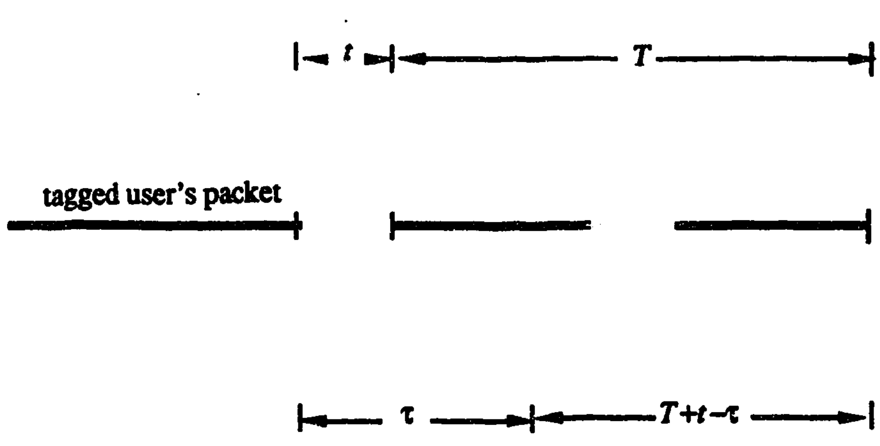


Figure 3.5 : Limits on X in an $A\rightarrow R$ transition

As in the $\bar{R} \rightarrow R$ transition

$$P^{(t)}(R,j,y_{aft} \mid A,i,y_{bef}) = \begin{cases} \sum_{n=0}^{K} P^{(n\tau+y_{aft},t)}(R,j,y_{aft} \mid A,i,y_{bef}) & 0 \le y_{aft} < \Delta \\ \sum_{n=0}^{K-1} P^{(n\tau+y_{aft},t)}(R,j,y_{aft} \mid A,i,y_{bef}) & \Delta \le y_{aft} < \tau \end{cases} \tag{3.12}$$

III. The tagged user is initially ready:

III.i Case $R \rightarrow \bar{R}$

This case is non-existent, since for the tagged user to become idle ($\bar{R}$) it must acquire the hub first. That is, the tagged user must go through an $R \rightarrow A$, then an $A \rightarrow \bar{R}$ transition. Therefore,

$$P(\bar{R},j,y_{aft} \mid R,i,y_{bef}) = 0 \tag{3.13}$$

III.ii Case $R \rightarrow A$

Since the model keeps track of the Y values of the tagged user, the value of the overhead period t is deterministic and given by $t=\tau-y_{bef}$, see Figure 3.6.

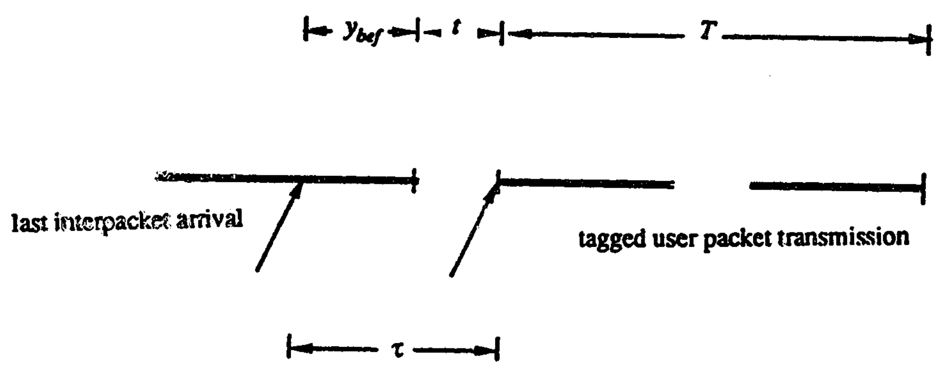


Figure 3.6 : Arrival of the tagged user's packet in an $R \rightarrow A$ transition

$$P(A,j,y_{aft} \mid R,i,y_{bef}) = \sum_{k=0}^{j-i} \sum_{l=0}^{j-k-i} \sum_{h=0}^{i} \frac{1}{k+l+h+1}$$

$$\Big\{ P_{\bar{L}}(t) \cdot P_{tx}(N-i-1,j-k-i,T) \cdot P_{idle}(k,N-j+k-1,t) \cdot P_{ready}(h,i,t) \cdot P_{new}(l,j-k-i,t)$$

$$+P_L(t) \cdot P_{tx}(N-i-1,j-k-1,T) \cdot P_{idle}(k,N-j+k,t) \cdot P_{ready}(h,i,t) \cdot P_{new}(l,j-k-i-1,t) \Big\} \qquad (3.14)$$

This is the same as the $\bar{R} \rightarrow A$ transition except for the term outside the summation, where it is dropped here, since the tagged user is already in the ready state.

III.iii Case $R \rightarrow R$

For a transition from y_{bef} to y_{aft}, the overhead period t is deterministic, see Figure 3.7, and is given by

$$t = \tau - y_{bef} + X - T$$

where

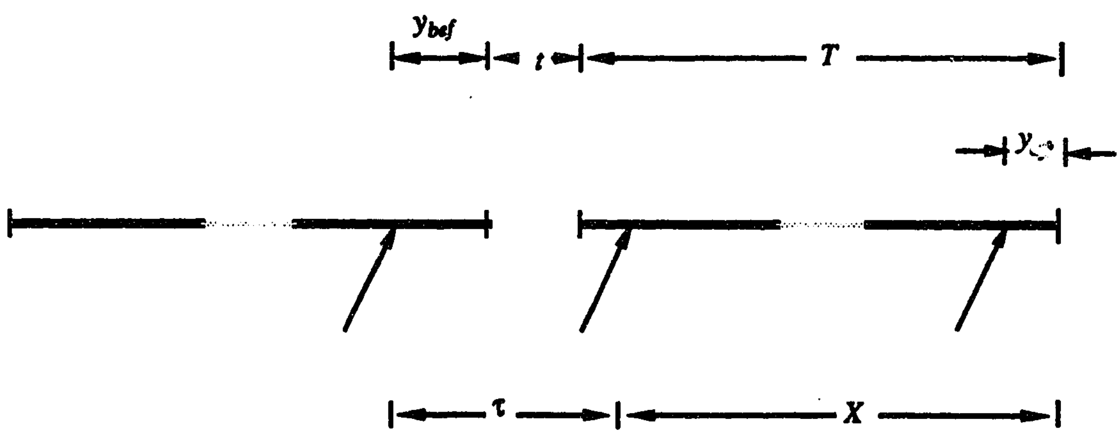


Figure 3.7 : Arrival of the tagged user's packet in an $R \rightarrow R$ transition

$$X = \begin{cases} K\tau + y_{aft} & 0 \le y_{aft} \le \Delta \\ (K-1)\tau + y_{aft} & \Delta \le y_{aft} < \tau \end{cases} \tag{3.15}$$

Therefore,

$$P(R,j,y_{aft} \mid R,i,y_{bef}) = \sum_{k=0}^{j-i+1} \sum_{l=0}^{j-k-i+1} \sum_{h=0}^{i}$$

$$\Big\{ P_{\bar{L}}(t) \cdot P_{tx}(N-i-1, j-k-i+1, T) \cdot P_{idle}(k, N-j+k-2, t) \cdot P_{ready}(h, i, t) \cdot P_{new}(l, j-k-i+1, t)$$

$$+ P_L(t) \cdot P_{tx}(N-i-1, j-k-i, T) \cdot P_{idle}(k, N-j+k-1, t) \cdot P_{ready}(h, i, t) \cdot P_{new}(l, j-k-i, t) \Big\} \tag{3.16}$$

This is the same as the $\bar{R} \to \bar{R}$ transition except that the first term in (3.1) is dropped here.

To solve for the steady state probabilities, a transformation from the three-dimensional state space to a one dimensional state space has to be employed. One such transformation may result in a transition probability matrix that has the form shown below.

The mapping operator, $\Phi\{\cdot\}$, is employed in such a transformation such that

$$\Phi\Big\{(\bar{R},i,0)\Big\} = i$$

$$\Phi\Big\{(A,i,0)\Big\} = N + i$$

$$\Phi\Big\{(R,i,Y)\Big\} = 2 \cdot N + 1 + i \cdot g + Y$$

The state space consists of $2N+1+g \cdot N$ states, which is $O(N \cdot g)$.

$$
\begin{array}{ll}
y_{bef}=0 & i=0 \\
 & i=1 \\
 & \vdots \\
 & i=N-1 \\
y_{bef}=0 & i=0 \\
 & i=1 \\
 & \vdots \\
 & i=N \\
y_{bef}=0 & i=0 \\
 & i=1 \\
 & \vdots \\
 & i=N-1 \\
y_{bef}=1 & i=0 \\
 & i=1 \\
 & \vdots \\
 & i=N-1 \\
\vdots & \\
y_{bef}=\tau-1 & i=0 \\
 & i=1 \\
 & \vdots \\
 & i=N-1
\end{array}
\left[
\begin{array}{c|c|c}
\bar{R}\rightarrow\bar{R} & \bar{R}\rightarrow A & \bar{R}\rightarrow R \\
\hline
A\rightarrow\bar{R} & A\rightarrow A & A\rightarrow R \\
\hline
R\rightarrow\bar{R} & R\rightarrow A & R\rightarrow R
\end{array}
\right]
$$

3.3. Model Analysis:

3.3.1 Elementary Analysis:

It is a simple matter to show that the above system is ergodic. Therefore, there exists a unique solution to the steady state equations in (3.17) below

$$\vec{\pi} = \vec{\pi} \cdot \mathbf{P} \quad \text{and} \quad \vec{\pi} \cdot \vec{1} = 1 \tag{3.17}$$

where,

P is the transition probability matrix given in the previous section.

$\vec{\pi}$ is a row vector containing the steady state probabilities.

$\vec{1}$ is a column vector of 1 elements.

Having obtained the steady state probabilities, they can be used in obtaining the system utilization and average packet delay. The system utilization U is given by

$$U = \frac{T}{\overline{O}+T} \tag{3.18}$$

where, $\overline{O}$ is the average overhead period.

Define P(A) as the steady state probability that the hub is acquired by the tagged user. Note that this is exactly equal to the proportion of hub acquisitions by the tagged user. Then, the utilization of the hub by the tagged user U_{tag} can be obtained and is given by

$$U_{tag} = P(A).U \tag{3.19}$$

where,

$$P(A) = \sum_{i=0}^{N} \pi_{(A,i,0)} \tag{3.20}$$

and $\pi_{(A,i,0)}$ is the steady state probability that the tagged user acquires the hub when there are i carry-over ready users at the start of the hub acquisition period. Notice that under steady operation, and in a symmetric system with $N+1$ users (including the tagged user), P(A) could be also given by

$$P(A)=\frac{1}{N+1}$$

The average delay of the tagged user is then given by:

$$\bar{D}=\frac{T}{U_{tag}}-\frac{1}{\sigma} \tag{3.21}$$

All that remains now is to find the average overhead period, $\bar{O}$, which can be obtained straightforwardly from the steady state probability distribution, and the transition probabilities as follows

$$\bar{O}=\sum_{i=0}^{N-1}\pi_{(\bar{R},i,0)}\times$$

$$\left\{\sum_{j=0}^{N}\sum_{y_{aft}=0}^{\tau-1}\sum_{t=1}^{\tau}t\cdot\left[P^{(t)}(\bar{R},j,0|\bar{R},i,0)+P^{(t)}(A,j,0|\bar{R},i,0)+P^{(t)}(R,j,y_{aft}|\bar{R},i,0)\right]\right.$$

$$\left.+\sum_{\substack{j=0\\i=0}}^{N}\sum_{y_{aft}=0}^{\tau-1}\sum_{t=\tau+1}^{\infty}t\cdot\left[P^{(t)}(\bar{R},j,0|\bar{R},i,0)+P^{(t)}(A,j,0|\bar{R},i,0)+P^{(t)}(R,j,y_{aft}|\bar{R},i,0)\right]\right\}$$

$$+\sum_{i=0}^{N}\pi_{(A,i,0)}\times$$

$$\left\{\sum_{j=0}^{N}\sum_{y_{aft}=0}^{\tau-1}\sum_{t=1}^{\tau}t\cdot\left[P^{(t)}(\bar{R},j,0|A,i,0)+P^{(t)}(A,j,0|A,i,0)+P^{(t)}(R,j,y_{aft}|A,i,0)\right]\right.$$

$$\left.+\sum_{\substack{j=0\\i=0}}^{N}\sum_{y_{aft}=0}^{\tau-1}\sum_{t=\tau+1}^{\infty}t\cdot\left[P^{(t)}(\bar{R},j,0|A,i,0)+P^{(t)}(A,j,0|A,i,0)+P^{(t)}(R,j,y_{aft}|A,i,0)\right]\right\}$$

$$+\sum_{i=0}^{N-1}\sum_{y_{bef}}^{\tau-1}\pi_{(R,i,y_{bef})}\times\sum_{j=0}^{N}\sum_{y_{aft}=0}^{\tau-1}\sum_{t=1}^{\tau}t\times$$

$$\left[P^{(t)}(\bar{R},j,0|R,i,y_{bef})+P^{(t)}(A,j,0|R,i,y_{bef})+P^{(t)}(R,j,y_{aft}|R,i,y_{bef})\right] \tag{3.22}$$

3.3.2 Enhanced Analysis:

The main approximation in the model described in section 3.2 is the treatment of carry-over ready users. A uniform distribution was used to approximate the distribution of the arrival instants of their transmission attempts at the hub. In this section, a better approximation for the distribution of carry-over ready users is presented.

Define $P_{tag}(t \mid R)$ as the probability that the tagged user arrives at the hub exactly in slot t after the hub release given that the tagged user was ready. The tagged user is ready if either an $R \rightarrow A$ or an $R \rightarrow R$ transition takes place. Therefore, $P_{tag}(t \mid R)$ is given by

$$P_{tag}(t \mid R) = \frac{1}{P_{tag}(R)} \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{N} \sum_{y_{aft}=0}^{\tau-1} \pi_{(R,i,y_{bef})}$$

$$[P(A,j,0 \mid R,i,y_{bef}) + P(R,j,y_{aft} \mid R,i,y_{bef})] \qquad (3.23)$$

where,

$$y_{bef} = \tau - t$$

and, $P_{tag}(R)$ is the steady state probability that the tagged use is ready and is given by

$$P_{tag}(R) = \sum_{i=0}^{N-1} \sum_{y_s=0}^{\tau-1} \pi_{(R,i,y_s)} \qquad (3.24)$$

Since the tagged user is modeled exactly, a better approximation in modeling carry-over ready users is to use $P_{tag}(t \mid R)$ rather than the uniform distribution used earlier. The model is therefore modified to employ an iterative method in which $P_{tag}(t \mid R)$ values are collected and fed back to the model to represent the distribution of carry-over ready users. As will be shown in the following section, such a method indeed yields more accurate results.

We can now redefine $P_{ready}(h,M,t)$ as

$$P_{ready}(h,M,t)=\begin{cases}\begin{bmatrix}M\\h\end{bmatrix} P_{tag}(t\mid R)^h\,(1-PF_{tag}(t\mid R))^{M-h} & 0\le h\le M \;\&\; t<\tau\\ P_{tag}(t\mid R)^M & h=M \;\&\; t=\tau\\ 0 & otherwise\end{cases}$$

where, $P_{tag}(t\mid R)$ is as defined above and $PF_{tag}(t\mid R)$ is the probability that a packet from a carry-over ready user arrives at the hub during time t from the hub release, and is given by

$$PF_{tag}(t\mid R)=\sum_{i=1}^{t}P_{tag}(i\mid R)$$

In the remainder of this section, the obvious differences between the distribution of the tagged user packet arrival time and the uniform distribution is shown and discussed.

Figure 3.8 shows the probability distribution function of the tagged user packet arrival, as obtained from the model described, for different retry time values (with g=20 and T=480 (bits)). Note how the distribution at values that are not factors of the packet length (100, 218) is very close to the uniform distribution. For the case of retry times that are factors of the packet length (120, 240), the distribution function is completely different and gives more mass to lower values of arrival times. The probability that the packet of the tagged user arrives at the hub at the 10th slot or earlier is 0.71, 0.47 and 0.5 for τ=120, τ=100 and the uniform distribution, respectively. Based on this, one might suspect that the average delay at τ=120 is lower than the average delay at τ=100. This, as was shown in

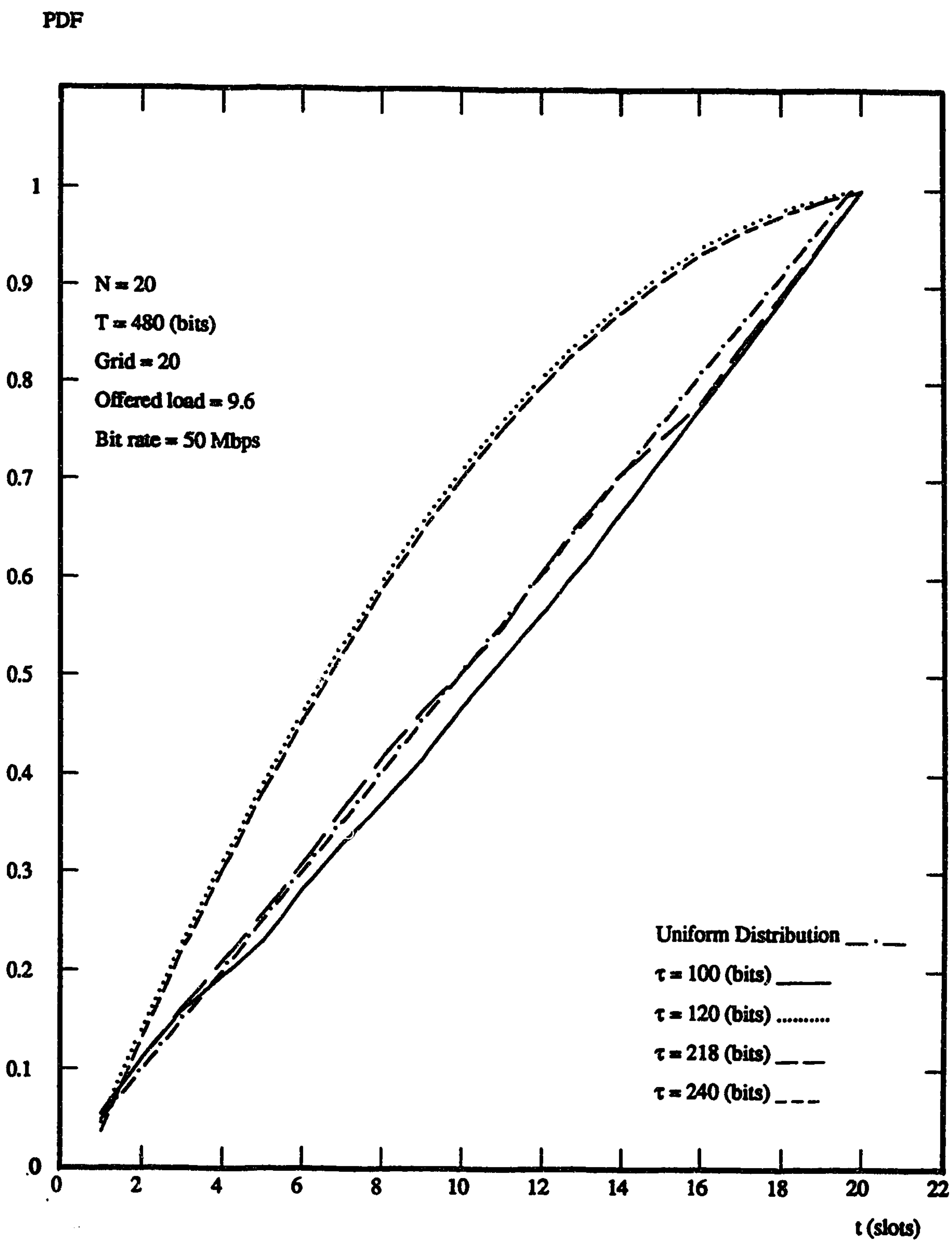


Figure 3.8: Distribution of tagged user packet arrival at different values of τ

Figure 3.2, is true, and as will be shown in the following section, is captured by the model.

The distribution of the tagged user's packet arrival is also dependent on the offered system load. Figure 3.9 shows the distribution of the tagged user packet arrival at different values of the offered load, at (T=480 bits, τ=80 bits -which is a factor of the packet length- and g=20). Note that only at light load is the distribution close to the uniform distribution. The results in Figure 3.9 suggest that the number of iterations required for steady state performance [16] increases with the offered load.

Finally, in Figure 3.10, the tagged user packet arrival distribution function is shown at different iterations of the model, at (T=480 bits, τ=120 bits -again a factor of the packet length- and g=20). Note that, after the first iteration, the distribution does not change significantly from one iteration to the next. This would imply that only a few iterations are required for convergence. Indeed, as will be shown in the following section, in most cases, four or less iterations are needed to obtain convergent and accurate results.

3.4. Numerical Results:

The delay throughput characteristics of CASLANs have been extensively studied in [42-45, 47-48]. In this section, emphasis is made on the effect of the retry time on the performance of CASLANs, which has been overlooked in all of these studies.

In Figure 3.11, analytical results after 6 iterations of the model versus the retry time are shown (with T=480 bits and N=20) at different offered loads. The results are compared to

16. The results of the model at iteration i are said to be at steady state if the results of iteration $i+1$ do not differ from those at iteration i.

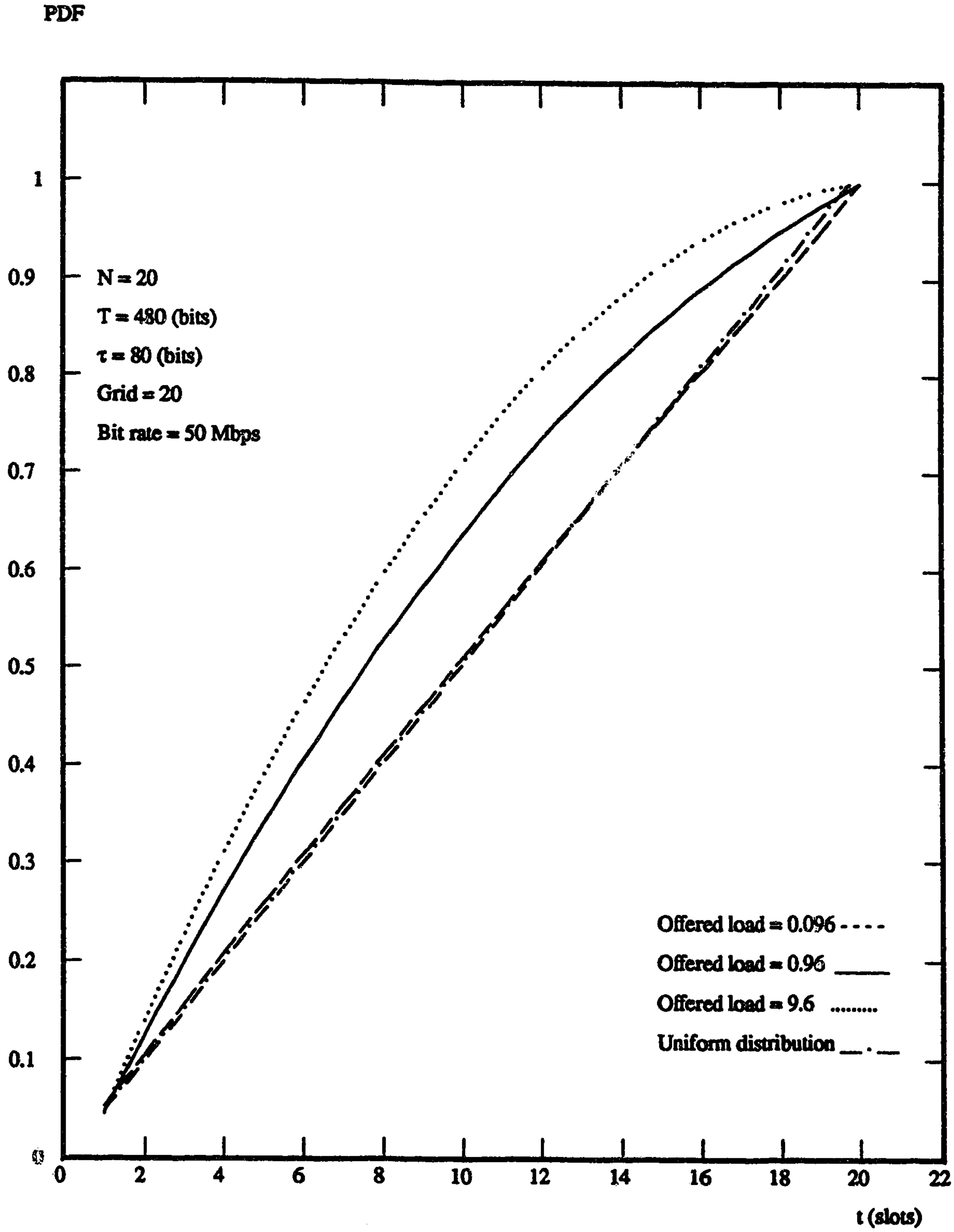


Figure 3.9: Distribution of tagged user packet arrival at different values of offered load

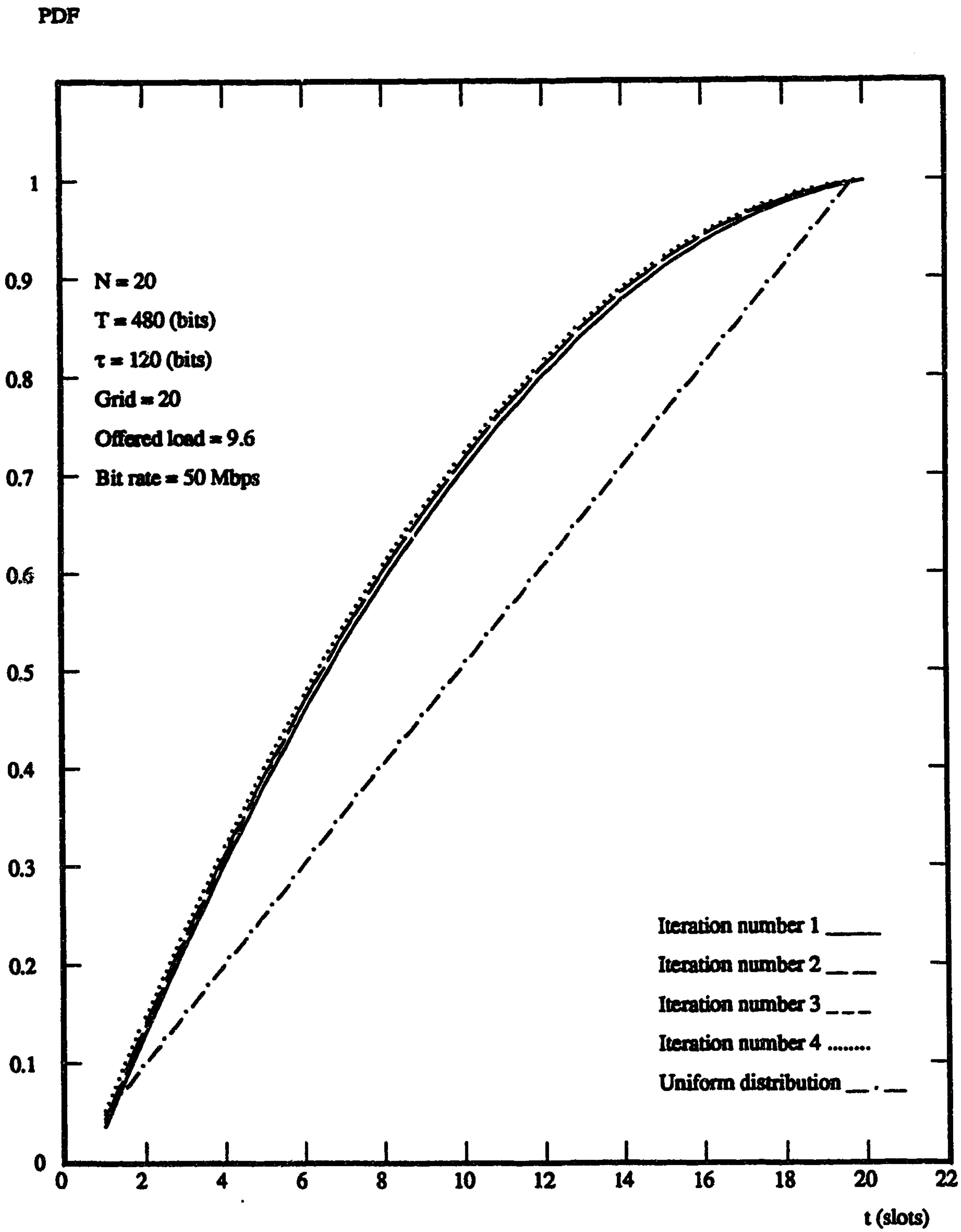


Figure 3.10: Distribution of tagged user packet arrival after 1,2,3 and 4 iterations

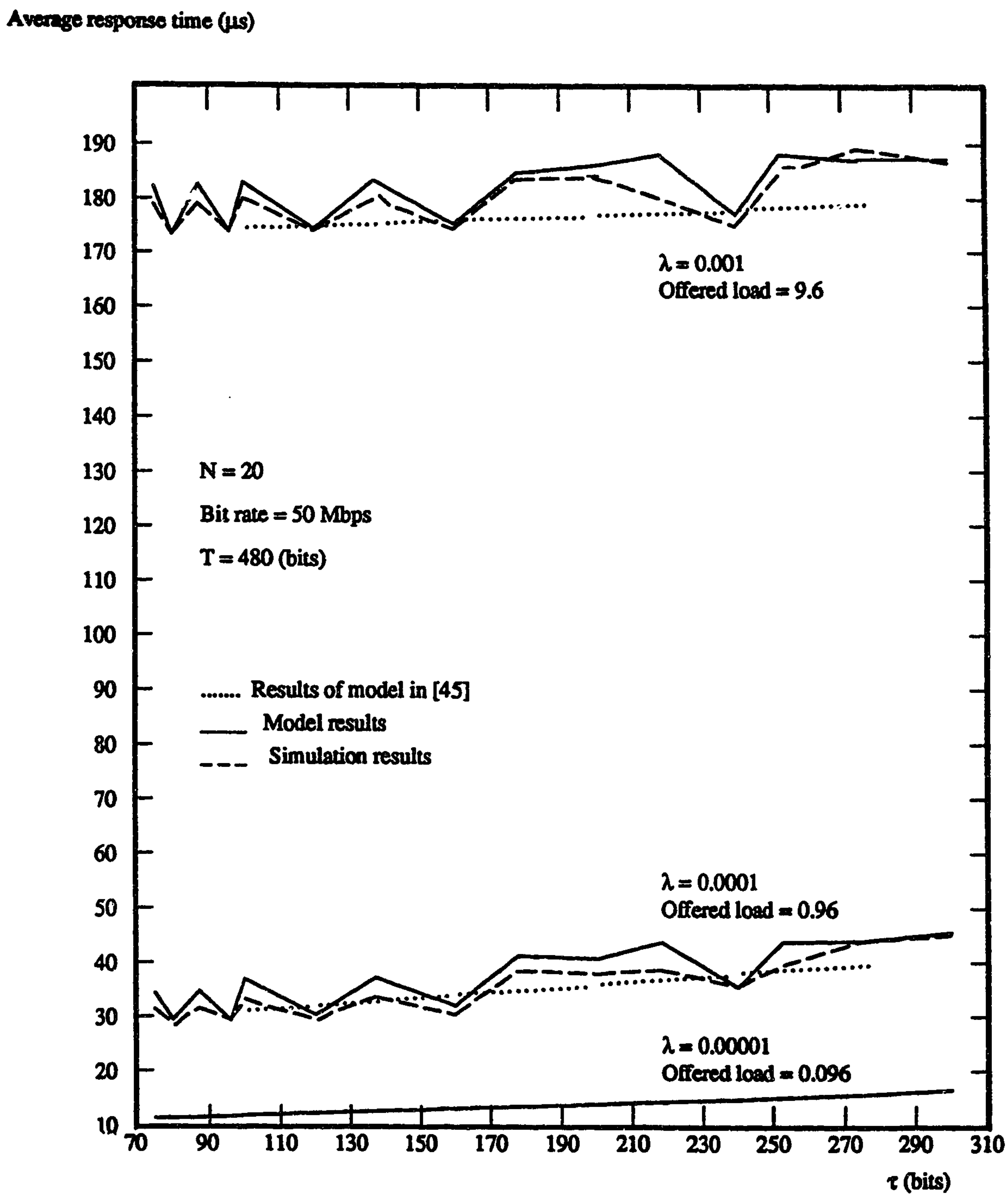


Figure 3.11: Response time at different values of τ in a 20 node system

those from simulation, as well as results of the model in [45]. The following observations can be made:

(1) At light load, results of the model conform to simulation results, and the higher the retry time the higher the delay.

(2) The phenomenon discovered in this thesis, and described in chapter 2, is captured by the model. That is, the results of the model show that the average delay at a retry time value that is a factor of the packet length, is better than at every higher retry time, and some lower retry time values. This is an exclusive merit of the model used here and is not present in any of the previous models [39-45].

(3) The accuracy of the model is very high. For instance, in Figure 3.11 at offered load=9.6, the error does not exceed 4.4%.

(4) Even though not shown here, the model still yields good results at higher retry time values. The accuracy at such retry time values, however, is not as good, but error percentages never exceed 10%.

The accuracy of the model is dependent on two main factors, namely:

1. The number of iterations used to obtain the results, and
2. the size of the grid, g.

As should be obvious to the reader, the larger the number of iterations and/or the grid size used, the more accurate is the model.

Figure 3.12 shows the average packet delay obtained from the model versus the retry time at different iterations (with N=20 nodes, T=480 and the offered load=9.6). As can be seen from Figure 3.12, the accuracy of the model increases with the number of iterations used. To better demonstrate this, in Table 3.2 average delay results of iterations 1 and 6 are listed along with simulation results, and the error percentages. The accuracy of the results at each and every point is improved with the use of the iterative method. Of special interest, are retry time values that are factors of the packet length. Note how at these retry time values the error almost vanish on the sixth iteration after being close to 12% on the first iteration. Also note that after the sixth iteration and at all retry time values other than 218 (bits), the error percentage is less than 2%. Even at this odd value, the error is about 4.4%.

Retry time	Simulation results	Results of iteration 1	Error percentage	Results of iteration 6	Error percantage
75	178.7	170	-4.9%	182.0	+1.8%
80	173.3	151	-12.9%	173.3	-0%
87	178.9	160.5	-10.6%	182.5	+2%
96	173.8	151.6	-12.8%	173.7	+0%
100	179.9	169	-6.1%	182.8	+1.6%
120	174	152.2	-12.5%	174.3	+0%
137	179.8	162	-9.9%	183.2	+1.9%
160	174.3	153.3	-12%	175.2	+0.1%
177	183.5	172	-6.3%	184.6	+0.1%
200	183.9	175	-4.8%	186.0	+1.1%
218	180	176	-7.7%	188.0	+4.4%
240	175	155	-11.4%	177.0	+1.1%
252	184.4	201	+9%	188.0	+1.9%
274	189	171	-9.5%	187.0	-1.1%
300	186.6	178	-10.5%	187.0	+0%

T = 480 bits; N = 20; Offered load = 9.6

Table 3.2: Average delay in (μs) at different retry time values versus simulation results.

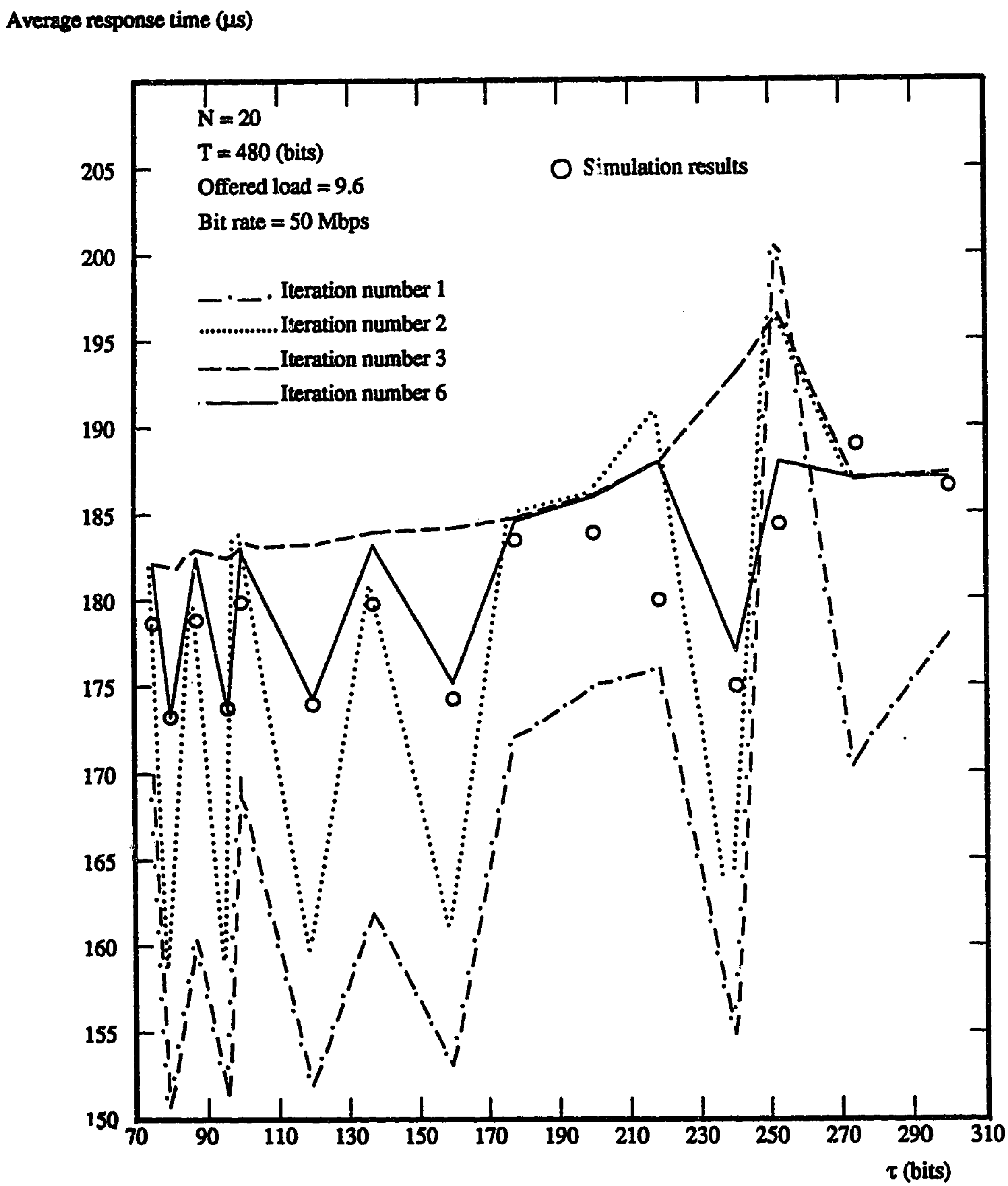


Figure 3.12: Response time at different values of τ in a 20 node system

The other factor affecting the accuracy of the model is the grid size. In Figure 3.13, average delay obtained after six iterations of the model is shown versus the retry time, at different grid sizes (with N=10 nodes, T=480 bits and variable offered loads). Simulation results are also presented for the sake of comparison. Note that the results are always more accurate with g=20, than with g=10. Increasing the grid size to values greater than 20 only increases the state space and does not increase the accuracy of the obtained results. Therefore, a grid size of 20 is sufficient to obtain reasonably accurate results.

The effect of the original distribution of carry-over ready users on the performance of the model are studied next. Recall that a uniform distribution was used as the initial distribution for carry-over ready users. In the following, it is shown that this initial distribution has little impact on the final results (obtained through the iterative method described above).

In Table 3.3, the probability density function of the arrival instant at the hub of the tagged user packet after different numbers of iterations is shown, using both the uniform and the truncated geometric as the original distribution of the carry-over ready users (with T=480 bits, τ=120 bits, N=20 and offered load=9.6). Note that regardless of the starting distributions, results after 6 iterations almost match. This was reflected in the final results, see Table 3.4. It is obvious from the results in Table 3.4 that regardless of the original distribution of carry-over ready users, the use of the iterative method yields similar results.

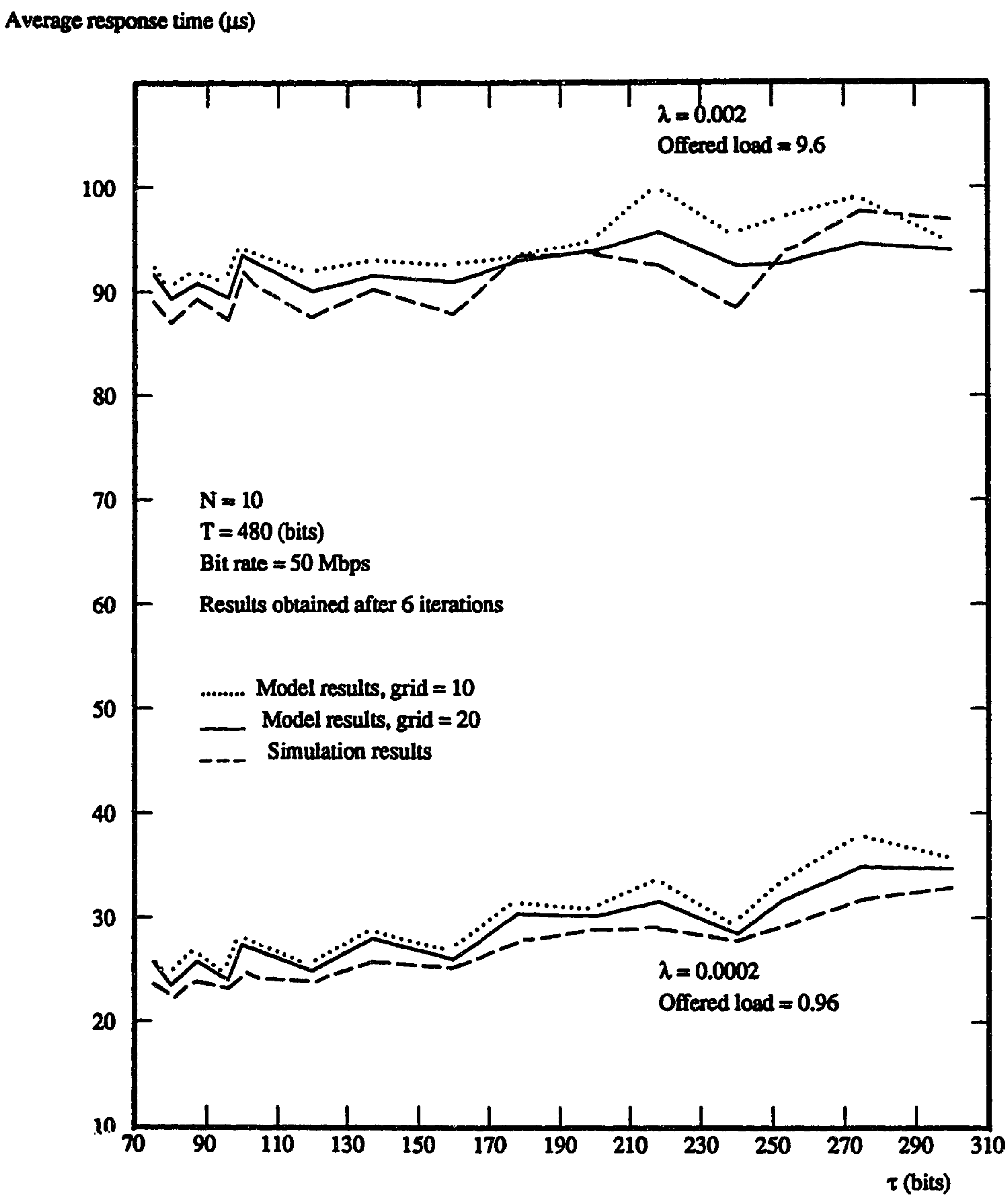


Figure 3.13: Response time at different values of τ in a 10 node system

t(slots)	Uniform distribution iteration #1	iteration #2	iteration #3	iteration #6	truncated geometric iteration #1	iteration #2	iteration #3	iteration #6
1	0.037530	0.045812	0.042593	0.043885	0.036708	0.046167	0.042453	0.043941
2	0.093918	0.094712	0.095090	0.094953	0.094178	0.094678	0.095105	0.094947
3	0.088945	0.089600	0.089954	0.089825	0.089180	0.089568	0.089968	0.089820
4	0.084001	0.084519	0.084849	0.084728	0.084212	0.084489	0.084862	0.084723
5	0.079087	0.079469	0.079774	0.079662	0.079273	0.079440	0.079786	0.079657
6	0.074202	0.074448	0.074730	0.074625	0.074364	0.074421	0.074741	0.074621
7	0.069347	0.069458	0.069716	0.069619	0.069485	0.069433	0.069726	0.069615
8	0.064520	0.064497	0.064732	0.064643	0.064634	0.064474	0.064741	0.064639
9	0.059723	0.059566	0.059777	0.059696	0.059813	0.059545	0.059786	0.059692
10	0.054953	0.054665	0.054853	0.054779	0.055020	0.054645	0.054861	0.054776
11	0.050213	0.049793	0.049958	0.049891	0.050256	0.049775	0.049965	0.049889
12	0.045500	0.044950	0.045092	0.045033	0.045520	0.044933	0.045098	0.045030
13	0.040815	0.040136	0.040255	0.040203	0.040812	0.040121	0.040260	0.040201
14	0.036158	0.035350	0.035446	0.035403	0.036132	0.035337	0.035451	0.035401
15	0.031527	0.030592	0.030666	0.030630	0.031478	0.030581	0.030670	0.030629
16	0.026922	0.025863	0.025914	0.025885	0.026851	0.025853	0.025917	0.025884
17	0.022344	0.021160	0.021189	0.021168	0.022250	0.021151	0.021192	0.021167
18	0.017793	0.016482	0.016490	0.016476	0.017677	0.016476	0.016492	0.016475
19	0.013278	0.011828	0.011814	0.011807	0.013135	0.011823	0.011815	0.011807
20	0.009222	0.007101	0.007108	0.007089	0.009021	0.007092	0.007110	0.007088

$N = 20$; $T = 480$ (bits); $\tau = 120$ (bits); offered load = 9.6

Table 3.3: The pdf of the arrival instant of the tagged user's packet with the original distribution of carry-over ready users being uniform or truncated geometric

retry time	iteration number	Uniform P(A)	U	D (μs)	Trancated geometric P(A)	U	D (μs)
100	1	.0516	.985	169.1	.0508	.985	171.8
	2	.0476	.985	184.5	.0487	.985	180
	3	.0480	.984	183.1	.0482	.984	182.3
	4	.0480	.984	183.1	.0480	.984	183
	5	.0480	.984	183.1	.0480	.984	183
	6	.0480	.984	183.1	.0480	.984	183
120	1	.0568	.981	152.2	.0533	.982	163
	2	.0543	.981	159.9	.0550	.981	158
	3	.0481	.962	184	.0478	.982	184
	4	.0501	.982	174.2	.0504	.983	173.8
	5	.0501	.982	174.3	.0498	.983	174.9
	6	.0501	.982	174.3	.0501	.983	174

$N = 20$; $T = 480$ (bits); offered load = 9.6

Table 3.4 : Average delay in (μs) at different retry time values with the original distribution of carry-over ready users being uniform or truncated geometric

Finally, heavy load performance results are compared to those obtained exactly in section 2.3 (for retry time values that are factors of the packet length). Table 3.5 shows results of the model at (N=10, T=480 bits and offered load=500) at different retry time value, that are factors of the packet length and compares them to the results in section 2.3. As noted from the table, the average delay results obtained using the model are very accurate as they almost match those results obtained in the exact analysis in section 2.3. -

retry time	Model results at offered load=500		Results of Exact model in section 2.3	
	D (μs)	U	D (μs)	U
60	97	.983	97.2	.987
80	97.4	.979	97.6	.983
96	97.6	.974	97.6	.980
120	98.2	.970	98.4	.975
160	99.5	.957	99.2	.967
240	101.2	.941	100.8	.952

Table 3.5: Model results at heavy load versus those in section 2.3.

3.5. Summary

In this chapter, a new CASLANs performance model was presented and analyzed. The model is based on following a tagged user in an exact analysis and approximating the behavior of the rest of the users by substituting the steady state behavior of the tagged user that is obtained through an iterative approach.

The model proved to provide very accurate results. As well, the model was able to capture the phenomenon described in chapter 2, namely that the average delay of CASLANs does not always increase by increasing the retry time. Retry time values that are factors of the packet length seem to yield local minima of the average delay. The excellent performance of the model is mainly attributed to the following:

(1) It does not assume independence between overhead periods, retry times and packet lengths, since it mimicks the exact behavior of the tagged user.

(2) The interpacket arrival times of packets form the tagged user are not dependent on the interpacket arrival times of other packets in particular. Indeed, it is only the length of the overhead period that affects the interpacket arrival times of packets from the tagged user.

(3) The model provides a better representation for ready users, since:

a) first-time ready users are modeled exactly, and

b) carry-over ready users are approximated through the use of the iterative approach.

The two major factors affecting the accuracy of the model are the grid size and the number of iterations used. It was shown that only a few iterations (typically four) are required to obtain steady state performance. It was also shown that a grid size of no more than 20 is sufficient to obtain very accurate results.

Heavy load results at retry times that are factors of the packet length were shown to be very accurate. In fact, the results almost matched those results obtained using the exact analysis in section 2.3. The latter analysis, however, is only restricted to the case where the retry time is a factor of the packet length.

Appendix 3.A

A Continuous-Time Performance Model For CASLANs

In this appendix, a continuous time CASLANs model is presented. This model is similar to the model in section 3.2 except that this one is continuous time. Hence, the model description is not repeated here. First, the following definitions are made.

- $$P_{tx}(M,n,T)=\begin{cases}\binom{M}{n}(1-e^{-\lambda T})^{n}(e^{-\lambda T})^{M-n} & 0\le n\le M\\ 0 & otherwise\end{cases}$$

  This is the probability of n transmissions from M idle nodes during the transmission time (T).

- $$P_{\bar{L}}=\begin{cases}e^{-\lambda(t+T-\tau)} & 0\le t\le\tau\\ e^{-\lambda T} & t>\tau\end{cases}$$

  This is the probability that the last node to release the hub will transmit during the next hub acquisition given that it occurred at time t.

- $P_L(t)=1-P_{\bar{L}}(t)$

- $f_a(i,t)=\dfrac{i(\tau-t)^{i-1}}{\tau^i}$

  This is the probability that the first of i carry-over ready users arrives at the hub at exactly time t. Since t, the overhead period, is defined once the hub is acquired, such a transmission would then be the first to arrive at the hub from any of the user groups (idle, first-time ready, carry-over ready or the tagged user). Subsequently, the packet would acquire the hub. In other words, this is the probability that one of i carry-over ready nodes would acquire the hub at exactly time t.

- $1-F_a(i,t)=\frac{(\tau-t)^i}{\tau^i}$

This is the probability that none of i carry-over ready nodes have a packet arrive at the hub before time t. Using the uniform distribution assumption for the remainder of the retry time after the release of the hub.

- $f_b(m,t)=\begin{cases} m\cdot a_K\cdot\lambda e^{-\lambda t}\cdot(1-a_K(1-e^{-\lambda t}))^{m-1} & 0<t\le\Delta \\ m\cdot b_K\cdot\lambda e^{-\lambda t}\cdot(1-a_K(1-e^{-\lambda(T-K\tau)})-b_K(e^{-\lambda(T-K\tau)}-e^{-\lambda t}))^{m-1} & \Delta<t\le\tau \end{cases}$

This is the probability that one out of m first-time ready nodes would acquire the hub at exactly time t, where

$$a_K=\frac{e^{\lambda(K+1)\tau}-1}{(1-e^{-\lambda\tau})(e^{\lambda T}-1)}$$

and

$$b_K=\frac{e^{\lambda K\tau}-1}{(1-e^{-\lambda\tau})(e^{\lambda T}-1)}$$

- $1-F_b(m,t)=\begin{cases} (1-a_K(1-e^{-\lambda t}))^m & 0<t\le\Delta \\ (1-a_K(1-e^{-\lambda(T-K\tau)})-b_K(e^{-\lambda(T-K\tau)}-e^{-\lambda t}))^m & \Delta<t\le\tau \end{cases}$

This is the probability that none of m first-time ready nodes have packets arriving at the hub during time t. Expressions for $f_b(m,t)$ and $F_b(m,t)$ were obtained in a fashion similar to that of appendix 3.B.

- $P_s(M,t)=\lambda M e^{-\lambda M t}$

  This is the probability that one out of M idle nodes acquires the hub at exactly time t.

- $P_{\bar{s}}(M,t)=e^{-\lambda M t}$

  This is the probability that none of the M idle users acquires the hub during time t.

Also, the following are defined

- $$P_{s'}(M,t)=\begin{cases} P_s(M-1,t) & 0<t\leq\tau \\ P_s(M-1,t)\cdot e^{-\lambda(t-\tau)}+P_{\bar{s}}(M-1,t)\cdot\lambda e^{-\lambda(t-\tau)} & t>\tau \end{cases}$$

  &

- $$P_{\bar{s}'}(M,t)=\begin{cases} P_{\bar{s}}(M-1,t) & 0<t\leq\tau \\ P_{\bar{s}}(M-1,t)\cdot e-\lambda(t-\tau) & t>\tau \end{cases}$$

Expressions for the transition probabilities joint with the value of t, $P^{(t)}(new,j,y_{aft}\mid old,i,y_{bef})$, are obtained first. Then an integration over t is performed. Please refer to section 3.2 for definitions of the dimensions of the Markovian chain.

Transition Probability Expressions

The transition probabilities provided here are similar to those in section 3.2 except that time is continuous. Explanation of the transition probability expressions is, therefore, not provided. The reader is encouraged to refer to section 3.2 for such an explanation.

I. Tagged user is initially not ready

<u>I.1 Case $\bar{R}\rightarrow\bar{R}$:</u>

The tagged user is not ready and will not become ready during the transition.

$$P^{(t)}(\bar{R},j,y_{aft}\mid\bar{R},i,y_{bef})=e^{-\lambda(t+T)}\cdot$$

$$\Big\{P_{\bar{L}}(t)\times\Big[P_{tx}(N-i-1,j-i,T)\times P_{s}(N-j-1,t)\times(1-F_a(i,t))\times(1-F_b(j-i,t))$$

$$+P_{tx}(N-i-1,j-i+1,T)\times P_{\bar{s}}(N-j-2,t)\times f_a(i,t))\times(1-F_b(j-i+1,t))$$

$$+P_{tx}(N-i-1,j-i+1,T)\times P_{\bar{s}}(N-j-2,t)\times(1-F_a(i,t))\times(1-F_b(j-i+1,t))\Big]$$

$$+P_{L}(t)\times\Big[P_{tx}(N-i-1,j-i-1,T)\times P_{s}(N-j,t)\times(1-F_a(i,t))\times(1-F_b(j-i-1,t))$$

$$+P_{tx}(N-i-1,j-i,T)\times P_{\bar{s}}(N-j-1,t)\times f_a(i,t))\times(1-F_b(j-i,t))$$

$$+P_{tx}(N-i-1,j-i,T)\times P_{\bar{s}}(N-j-1,t)\times(1-F_a(i,t))\times f_b(j-i,t))\Big]\Big\} \qquad (3.A.1)$$

<u>I.ii Case $\bar{R}\rightarrow A$</u>

The tagged user acquires the hub during the transition.

$$P^{(t)}(A,j,y_{aft}\mid\bar{R},i,y_{bef})=\lambda e^{-\lambda t}\times$$

$$\Big\{P_{\bar{L}}(t)\times P_{tx}(N-i-1,j-i,T)\times P_{\bar{s}}(N-j-1,t)\times(1-F_a(i,t))\times(1-F_b(j-i,t))$$

$$+P_{L}(t)\times P_{tx}(N-i-1,j-i-1,T)\times P_{\bar{s}}(N-j,t)\times(1-F_a(i,t))\times(1-F_b(j-i-1,t))\Big\} \qquad (3.A.2)$$

<u>I.iii Case $\bar{R}\rightarrow R$</u>

The tagged user becomes ready during the transition. Please refer to Figure 3.4.

$$P^{(X,t)}(R,j,y_{aft}\mid\bar{R},i,y_{bef})=\lambda e^{-\lambda(t+T-X)}\times$$

$$\Big\{P_{\bar{L}}(t)\times\Big[P_{tx}(N-i-1,j-i,T)\times P_s(N-j-1,t)\times(1-F_a(i,t))\times(1-F_b(j-i,t))$$

$$+P_{tx}(N-i-1,j-i+1,T)\times P_{\bar{s}}(N-j-2,t)\times f_a(i,t))\times(1-F_b(j-i+1,t))$$

$$+P_{tx}(N-i-1,j-i+1,T)\times P_{\bar{s}}(N-j-2,t)\times(1-F_a(i,t))\times(1-F_b(j-i+1,t))\Big]$$

$$+P_L(t)\times\Big[P_{tx}(N-i-1,j-i-1,T)\times P_s(N-j,t)\times(1-F_a(i,t))\times(1-F_b(j-i-1,t))$$

$$+P_{tx}(N-i-1,j-i,T)\times P_{\bar{s}}(N-j-1,t)\times f_a(i,t))\times(1-F_b(j-i,t))$$

$$+P_{tx}(N-i-1,j-i,T)\times P_{\bar{s}}(N-j-1,t)\times(1-F_a(i,t))\times f_b(j-i,t))\Big]\Big\} \qquad (3.A.3)$$

Recall that $X=n\tau+y_{aft}$. Summing over all values of X, then

$$P^{(t)}(R,j,y_{aft}\,|\,\bar{R},i,y_{bef})=\begin{cases}\sum_{n=0}^{K}P^{(n\tau+y_{aft},t)}(R,j,y_{aft}\,|\,\bar{R},i,y_{bef}) & 0<y_{aft}\le\Delta\\ \sum_{n=0}^{K-1}P^{(n\tau+y_{aft},t)}(R,j,y_{aft}\,|\,\bar{R},i,y_{bef}) & \Delta<y_{aft}\le\tau\end{cases} \qquad (3.A.4)$$

II. Tagged user is initially acquiring the hub

II. i Case $A\rightarrow\bar{R}$

The tagged would release the hub and does not become ready during the transition.

$$P^{(t)}(\bar{R},j,y_{aft}\,|\,A,i,y_{bef})=e^{-\lambda(t+T-\tau)}\times\Big[P_{tx}(N-i,j-i,T)\times P_s(N-j,t)\times(1-F_a(i,t))\times(1-F_b(j-i,t))$$

$$+P_{tx}(N-i,j-i+1,T)\times P_{\bar{s}}(N-j-1,t)\times f_a(i,t))\times(1-F_b(j-i+1,t))$$

$$+P_{tx}(N-i,j-i+1,T)\times P_{\bar{s}}(N-j-1,t)\times(1-F_a(i,t))\times f_b(j-i+1,t))\Big] \tag{3.A.5}$$

II. ii Case $A \to A$

For the tagged user to reacquire the hub, t must be greater than τ. Thus, there should be no ready users at the first embedding point.

$$P^{(t)}(A,j,y_{aft} \mid a,i,y_{bef}) = \lambda e^{-\lambda(t-\tau)} \times P_{tx}(N,0,T)\times P_{\bar{s}}(0,N,t) \tag{3.A.6}$$

II.iii Case $A \to R$

This case is similar to the $\bar{R} \to R$ case, except that the tagged user must wait for a time equal to τ before being able to contend for the hub.

$$P^{(X,t)}(R,j,y_{aft} \mid A,i,y_{bef}) = \lambda e^{-\lambda(t+T-\tau-X)} \times \Big[P_{tx}(N-i,j-i,T)\times P_s(N-j,t)\times(1-F_a(i,t))\times(1-F_b(j-i,t))$$

$$+P_{tx}(N-i,j-i+1,T)\times P_{\bar{s}}(N-j-1,t)\times f_a(i,t))\times(1-F_b(j-i+1,t))$$

$$+P_{tx}(N-i,j-i+1,T)\times P_{\bar{s}}(N-j-1,t)\times(1-F_a(i,t))\times f_b(j-i+1,t))\Big] \tag{3.A.7}$$

As in the $\bar{R} \to R$ case, and summing over all values of X, then

$$P^{(t)}(R,j,y_{aft} \mid A,i,y_{bef}) = \begin{cases} \sum_{n=0}^{K} P^{(n\tau+y_{aft},t)}(R,j,y_{aft} \mid A,i,y_{bef}) & 0<y_{aft}\le\Delta \\ \sum_{n=0}^{K-1} P^{(n\tau+y_{aft},t)}(R,j,y_{aft} \mid A,i,y_{bef}) & \Delta<y_{aft}\le\tau \end{cases} \tag{3.A.8}$$

III. Tagged user is initially ready

III.i Case $R \to \bar{R}$

This case is non-existent, since for the tagged user to become idle it must acquire the hub first. That is, go through on $R \to A$, then $A \to \bar{R}$ transitions. Therefore,

$$P(\bar{R},j,y_{aft}\mid R,i,y_{bef})=0 \tag{3.A.9}$$

III. ii Case $R \to A$

In this case, the overhead period is deterministic and is given $t=\tau-y_{bef}$. (Refer to figure 3.6)

$$P(A,j,y_{aft}\mid R,i,y_{bef})=P_{\bar{L}}(t)\times P_{tx}(N-i-1,j-i,T)\times P_{\bar{s}}(N-j-1,t)\times(1-F_a(i,t))\times(1-F_b(j-i,t))$$

$$+P_L(t)\times P_{tx}(N-i-1,j-i-1,T)\times P_{\bar{s}}(N-j,t)\times(1-F_a(i,t))\times(1-F_b(j-i-1,t)) \tag{3.A.10}$$

III.iii Case $R \to R$

Refer to figure 3.7. The overhead period t is deterministic and is given by

$$t=\tau-y_{bef}+X-T$$

where,

$$X=\begin{cases}K\tau+y_{aft} & 0<y_{aft}\le\Delta\\ (K-1)\tau+y_{aft} & \Delta<y_{aft}\le\tau\end{cases}$$

Therefore,

$$P(R,j,y_{aft}\mid R,i,y_{bef})=P_{\bar{L}}(t)\times\Big[P_{tx}(N-i-1,j-i,T)\times P_s(N-j-1,t)\times(1-F_a(i,t))\times(1-F_b(j-i,t))$$

$$+P_{tx}(N-i-1,j-i+1,T)\times P_{\bar{s}}(N-j-2,t)\times f_a(i,t))\times(1-F_b(j-i+1,t))$$

$$+P_{tx}(N-i-1,j-i+1,T)\times P_{\bar{s}}(N-j-2,t)\times(1-F_a(i,t))\times(1-F_b(j-i+1,t))\Big]$$

$$+P_L(t)\times\Big[P_{tx}(N-i-1,j-i-1,T)\times P_s(N-j,t)\times(1-F_a(i,t))\times(1-F_b(j-i-1,t))$$

$$+P_{tx}(N-i-1,j-i,T)\times P_{\bar{s}}(N-j-1,t)\times f_a(i,t))\times(1-F_b(j-i,t))$$

$$+P_{tx}(N-i-1,j-i,T)\times P_{\bar{s}}(N-j-1,t)\times(1-F_a(i,t))\times f_b(j-i,t))\Big] \tag{3.A.11}$$

After obtaining the transition probabilities and the steady state probabilities, the average delay could be obtained through solving the set of integral equations in (3.A.12) below. Refer to Figure 3.14.

$$\bar{D} = Pr(A\,|\,\bar{R}) \times P(\bar{R}) \times (T+\tau) + \sum_{i=0}^{N-1}\left[\int_{X_1|i} X_1\, Pr(R,X_1\,|\,\bar{R},i,0)\times Pr(\bar{R},i,0).dX_1 \right.$$

$$\left. + \sum_{j=i-1}^{N-1}\int_{X|i} d_{X,j}\times Pr(R,j,X\,|\,\bar{R},i,0)\times Pr(\bar{R},i,0)\,.\,dX \right] \qquad (3.A.12a)$$

Where,

$$d_{X,j} = \bar{t}(X,j) + T + \sum_{k=j-1}^{N-1}\int_{s|j} d_{S,k}\times Pr(R,k,S\,|\,R,j,X)\,.\,dS + \sum_{k=j}^{N-1}\int_{S} Pr(A,k,T\,|\,R,j,X)\,.\,dS \qquad (3.A.12b)$$

and $P(\bar{R})$ = Probability of not ready

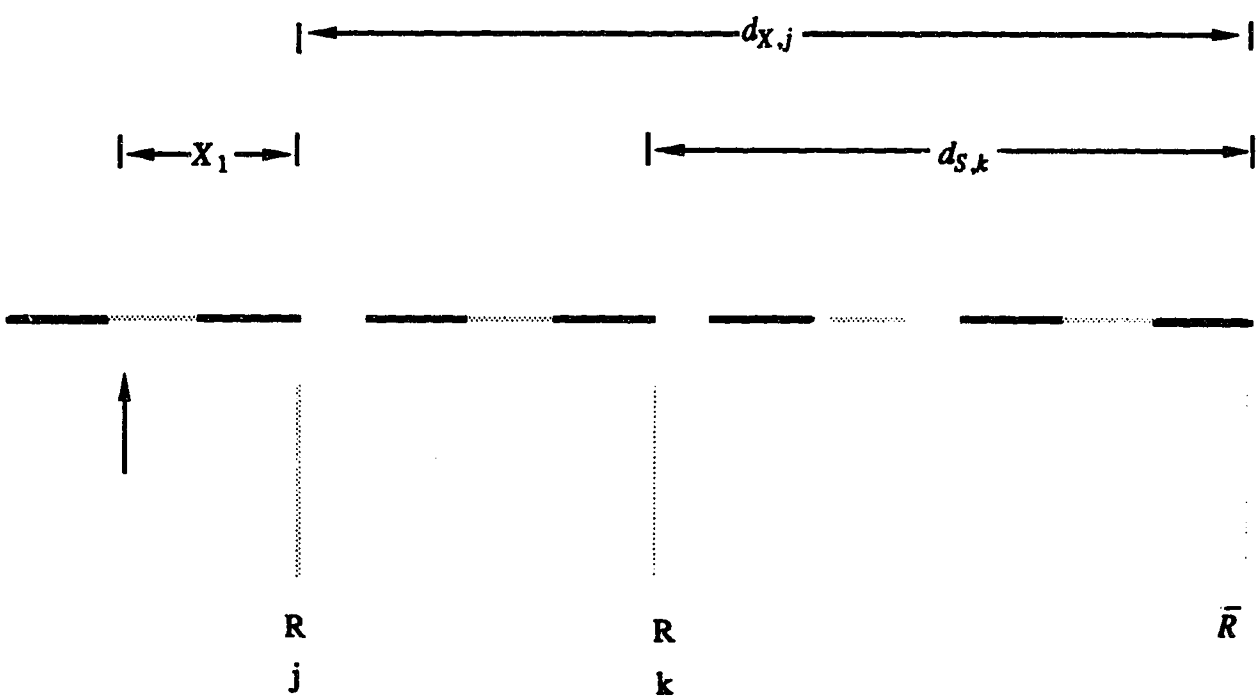


Figure 3.14: Delay in CASLANS.

Appendix 3.B

Derivation of $P_a(t)$

In this appendix, the probability that a first-time ready user contends for the hub at exactly slot t is derived.

Refer to Figure 3.15, let $P_{arr}(Z_1)$ be the probability that the first interpacket arrival time of a packet is Z_1. $P_{arr}(Z_1)$ is given by

$$P_{arr}(Z_1) = \sigma(1-\sigma)^{Z_1-1} \tag{3.B.1}$$

The second interpacket arrival time would be at a time which is $i \cdot \tau$ later, where $1 \le i \le K$, and is dependent on the overhead period, t. Let W be the time of arrival of the packet, measured from the last hub release, of the packet on its second interarrival. Thus $W = Z_2 + t$, see Figure 3.15. We make a distinction between two cases,

1) $0 < W \le \tau - \Delta$: In this case, $Z_1 = W + \Delta + i\tau$, where $0 \le i \le K-1$ (Figure 3.15a).

2) $\tau - \Delta < W \le \tau$: In this case, $Z_1 = W - \tau + \Delta + i\tau$, where $0 \le i \le K$ (Figure 3.15b).

Define $P_a(W)$ as the probability that a packet from a new user arrives at or after time W from the last hub release. $P_a(W)$ is given by

$$P_a(W)=\begin{cases}\sum_{i=0}^{K-1}\frac{\sigma(1-\sigma)^{W+\Delta+i\tau-1}}{1-(1-\sigma)^T} & 0<W\le\tau-\Delta\\ \sum_{i=0}^{K}\frac{\sigma(1-\sigma)^{W+\Delta+(i-1)\tau-1}}{1-(1-\sigma)^T} & \tau-\Delta<W\le\tau\end{cases} \tag{3.B.2}$$

Replacing the parameter W by t and simplifying the sum in (B.2) above, the following expression is obtained.

$$P_a(t)=\begin{cases}\frac{\sigma(1-\sigma)^{t+\Delta-1}\cdot(1-(1-\sigma)^{K\tau})}{(1-(1-\sigma)^T)\cdot(1-(1-\sigma)^{\tau})} & 0<t\le\tau-\Delta\\ \frac{\sigma(1-\sigma)^{t+\Delta-\tau-1}\cdot(1-(1-\sigma)^{(K+1)\tau})}{(1-(1-\sigma)^T)\cdot(1-(1-\sigma)^{\tau})} & \tau-\Delta<t\le\tau\end{cases} \tag{3.B.3}$$

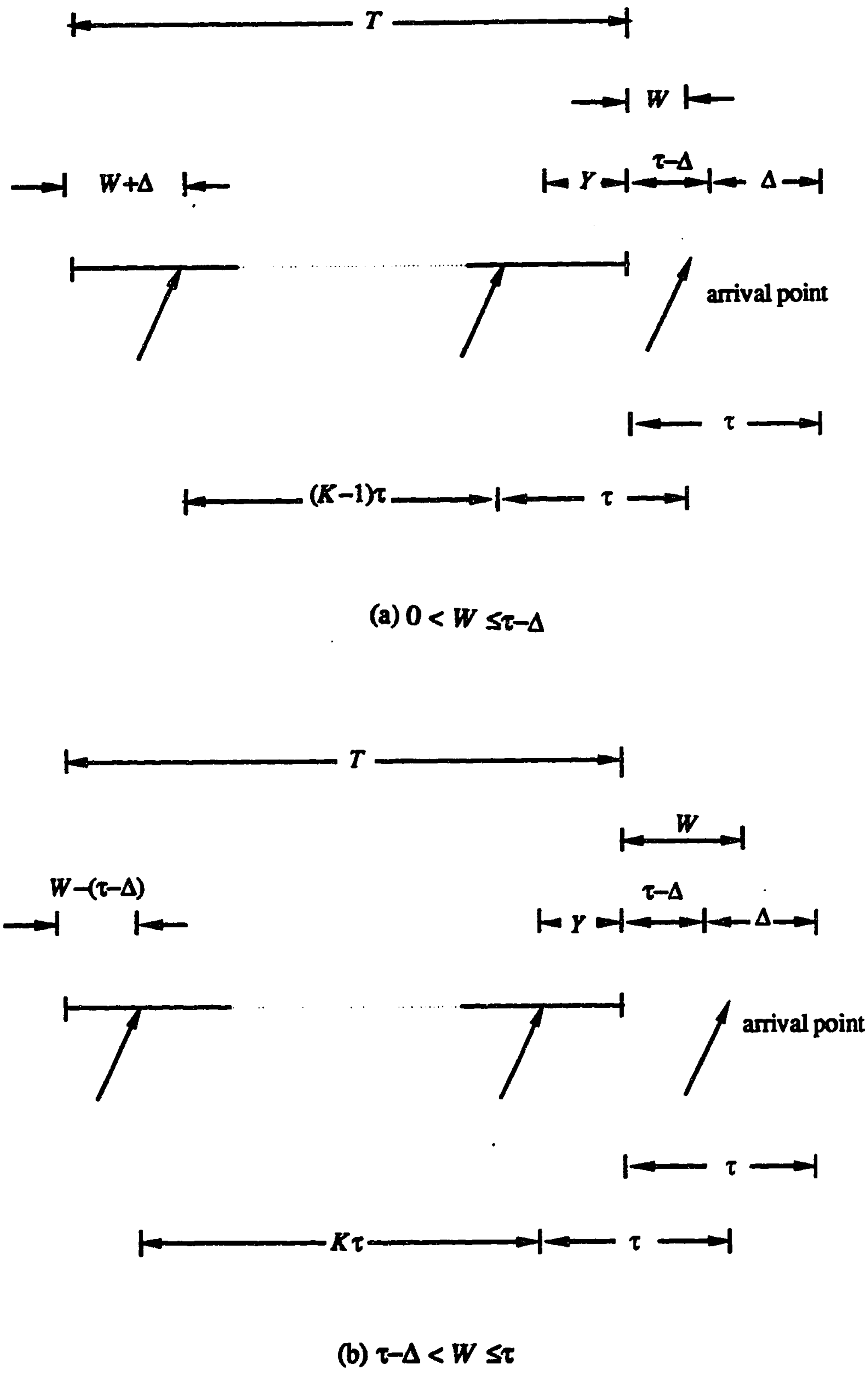


Figure 3.15: The second interpacket arrival of a new user

Chapter 4

CASLANs in a Real-Time Environment

In chapters 2 and 3, new performance and behavioral characteristics of CASLANs were shown, analyzed and modeled. Some of these characteristics seem to suggest that CASLANs might perform very well under time constraints. Specifically, the guaranteed packet delivery and the bounded delay, at heavy load, at retry times that are factors of the packet length, are useful in this respect. This chapter studies the performance of CASLANs in real-time (time constrained) environments. Both soft and hard real-time systems are considered. For soft real-time systems, a simulation model is developed from which performance results are obtained. For hard real-time CASLANs, a performance model, that is an extension of the model presented in chapter 3, is introduced and analyzed.

4.1. Background

In real-time applications [50], time is the most important resource to manage. Tasks must be assigned and scheduled in such a way that they can be completed before their corresponding deadlines expire. A task which is not completed before its deadline, is considered useless. Applications of this nature include: on-board flight controllers, process control systems and automated manufacturing plants.

Recently, there has been an increasing interest in supporting real-time communication applications such as packetized voice [51-54] and real-time control [55]. A real-time communication system is said to be *hard* if packets that are known to have exceeded their *laxi-*

ties[17] are removed from the system. Otherwise, it is a *soft* real-time communication system.

The communication requirements of such real-time applications differ from those of traditional non-real-time multiple access communication in that:

(1) A certain amount of packet loss is usually tolerable, see section 5.5.

(2) A packet which is not successfully transmitted before its deadline is considered lost, regardless of whether or not it is eventually delivered at the receiving end.

(3) Reliability is crucial, since failures would result in delayed successful packet transmissions.

(4) The primary performance objective is no longer to minimize average delays, but rather to maximize the percentage of successfully transmitted packets (within their allowable deadlines).

(5) The distribution of packet delay, and no longer its mean value, is the principal performance measure.

(6) In many real-time multiple access protocols scheduling is usually employed to reduce the probability of packets lost.

Classical real-time scheduling techniques [56-58] include:

(1) Shortest length packet first: The rationale here is that by sending the shortest packets first, more packets could be sent (as short packets do not use much of the

17. The laxity of a packet is defined as the maximum allowable time before sending the packet without having that packet exceed its deadline.

channel bandwidth). This technique is the same as shortest job first.

(2) Minimum laxity first: In this scheme, packets with the least laxity are given priority. The rationale here is to prevent such packets from being lost.

(3) Minimum deadline first: Where the deadline is the packet transmission time plus the packet laxity.

(4) Random order: Packets are transmitted at random. In the literal sense, this is equivalent to "no scheduling".

(5) Fixed priority: Some packets are assumed more important than others, and are thus transmitted first.

(6) First-Come-First-Served (FCFS): This method is mostly preferred for its fairness.

(7) Last-Come-First-Served (LCFS): That is, the last packet to arrive at a station is transmitted first. The rationale here is that, rather than lose the last arriving packet by trying to transmit older packets, where they may be lost themselves, transmit it to guarantee its success.

In multiple access communications channels, two scheduling problems exist:

(1) Scheduling must be performed in a distributed fashion. A global scheduling policy must be adopted by all stations. Therefore, a scheduling policy must be employed using mostly local information, or information extracted from the channel that may be already too old when extracted.

(2) Since no station has knowledge of other packets in the system, no scheduling policy would guarantee all packets to be successfully transmitted.

4.2. Previous Work

There is no published work on the use of CASLANs (or any variation of CASLANs) in a real-time environment. However, many multiple access protocols have been studied and others proposed for real-time applications.

(a) Controlled access protocols:

Fixed assignment schemes have not received much consideration for use in real-time communication. This is because in fixed assignment schemes, channel bandwidth is wasted by stations with no packets to transmit, while other stations that have packets with deadlines are waiting their turn, thus resulting in packet loss. However, variations of TDMA exist for integrated voice and data communications [59, 60].

Two characteristics of demand assignment protocols make them more attractive for real time applications.

(1) As in fixed assignment protocols, a guaranteed amount of the channel capacity is provided for every station (upon request.)

(2) Token passing or reservation posting times are usually small compared to the packet transmission time. Therefore, unlike fixed assignment protocols, channel bandwidth is not wasted by stations with nothing to transmit.

Santos et al [61] performed a comparative evaluation of round robin and minimum-laxity first scheduling on the token ring [5] and the token bus [6] networks. The results showed that if packet laxities exceed NT (N is the number of stations and T is the packet transmission time), then round robin scheduling results in a lower loss probability.

Earlier, Kim [62] showed that by just discarding packets that exceeded their laxities in the token ring protocol, lower probabilities of loss could be achieved.

Ramamrithan [63] studied the problem of computing bounds on packet delays in hard real-time demand assignment protocols. Based on the developed model, Ramamrithan computed the minimum packet interarrival times needed to guarantee a certain maximum delay.

(b) Random access protocols:

By far, random access protocols are the most explored multiple access communication class for real-time communication. Advantages of random access protocols for real-time communication include:

(1) No bandwidth is wasted by a station with nothing to transmit (as in fixed assignment protocols) or in token passing and reservation posting (as in demand assignment protocols).

(2) The packet delays are more dependent on the offered load than the number of stations connected to the channel.

The major disadvantage of random access protocols, however, is the variable packet delay which, in many protocols, is unbounded.

Many random access protocols were studied for real-time applications [64-74]. In [64] and [65], the performance of Ethernet [17] for real time applications was studied. It was found that for values of channel utilization less than 1/2, the probability of loss is very small. Variations of the CSMA/CD protocol for voice or combined voice and data transmission were suggested [66-72] and are reported in section 5.1.

Kurose et al [73] proposed a generalization of the window protocol using three scheduling policies: FCFS, LCFS, and random. This was easily done by selecting the first half, second half or a half at random when a collision occurs, respectively. It was found that a LCFS policy is better than FCFS and random service order in terms of the probability of loss. The authors also proposed a hard real-time protocol, with a FCFS scheduling, where a packet is discarded if its waiting time exceeded its laxity. The results showed that such a protocol outperforms all three soft real-time protocols above.

Zhao and Ramamrithan [74], in a simulation study, proposed four modifications to the VT-CSMA protocol [18]. This was achieved by using different packet parameters in conjunction with the virtual clock to transmit packets. These parameters, which were reflected into scheduling policies, were: the packet arrival time, the packet length and the packet laxity. The results show that, a shortest-packet first and a minimum-deadline first policy outperforms the FCFS and the minimum-laxity first policies in terms of percentage packet loss. This is because the former policies are biased towards short packets.

4.3. Performance Evaluation of Time-Constrained CASLANs

In this section the performance of CASLANs in real-time environment is studied and analyzed. Both soft and hard real time systems are considered (see section 4.3.1 and 4.3.2, respectively). Because of the interesting behavioral characteristics of fixed-packet-length CASLANs, demonstrated in chapter 2, emphasis will be on such a case.

In any real-time, error-free, communication network, packet loss is attributed to two sources: *blocking loss* and *channel access loss*. The blocking loss is due to packets arriving at a node with full buffers and are, therefore, blocked. On the other hand, channel access loss is due to packets exceeding their laxities while waiting to access the channel. In this chapter, only channel access loss is considered. It is, therefore, assumed that the packet laxity and packet deadline are defined from the moment a packet reaches the top of the queue, and is waiting to access the channel. For single buffer systems, such times start from the moment of the packet arrival at a node.

For soft real-time systems, the *packet survival function,* defined as the percentage of packets with delays exceeding a certain given delay value, is used as the performance measure. For hard real-time systems, the probability of packet lost, defined as the percentage of packets that are not transmitted before their laxities expire, is used as the performance measure.

4.3.1. Soft Real-Time CASLANs

For soft real-time systems, the probabilities of loss can be computed analytically by use of the packet delay distribution. For instance, if the deadline to receive a packet is DL, the probability of loss is given by $\Pr(D > DL)$, where D is the packet delay.

For CASLANs, deriving the delay distribution seems to be an extremely complex and involved process. Indeed, such a process would involve finding the inverse Laplace transform of the moment generating function of the delay expression given by the set of equations in 3.A.12. Therefore, simulation is used to study the performance of soft real-time CASLANs.

The simulation model used here has the same assumptions as the one described in section 2.1. The packet survival function is used as the performance measure.

Figures 4.1-4.4 show the packet survival function for CASLANs with different retry time values at different offered load values (with N=20 nodes, T=480 bits and a bit rate of 50 Mbps). The following observations are made:

(1) The packet survival rate of CASLANs is acceptable over all offered load ranges. For instance, at light load, an average of 96% of packets survive a deadline of 20 μs (an average of only two retry attempts). At higher loads, the performance is still very reasonable. For example, at heavy load (offered load = 9.6) and with the given parameters, a deadline of 1 ms almost guarantees no packet loss.[18]

(2) In most cases, the packet survival rate decreases with increasing the retry time (see (3) below) and/or the offered load.

(3) At high to heavy loads increasing the retry time does not always mean a decrease

18. Such a performance far exceeds the requirements for voice packet transmission, see section 5.5.

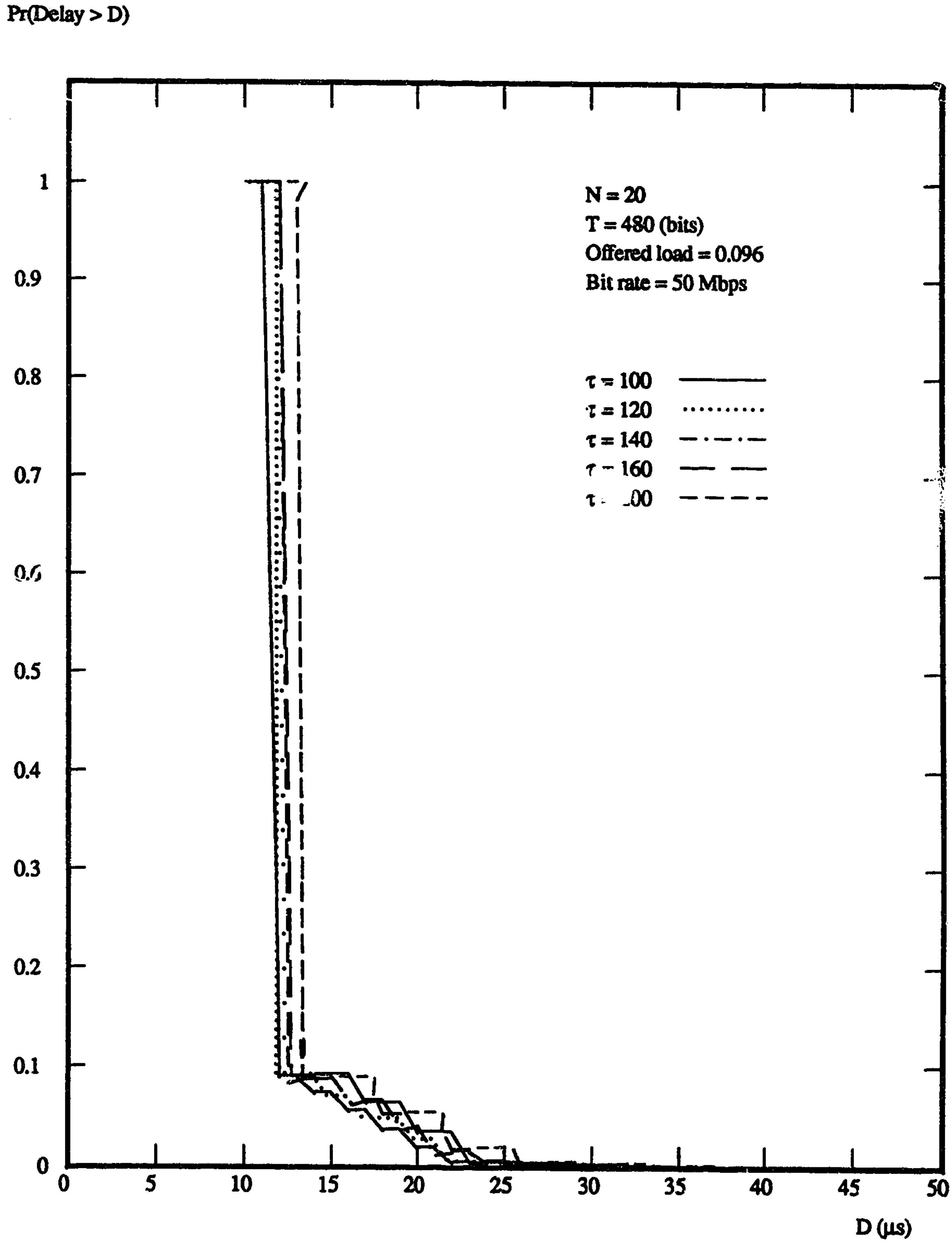


Figure 4.1: Survivor function of CASLANs at different values of τ

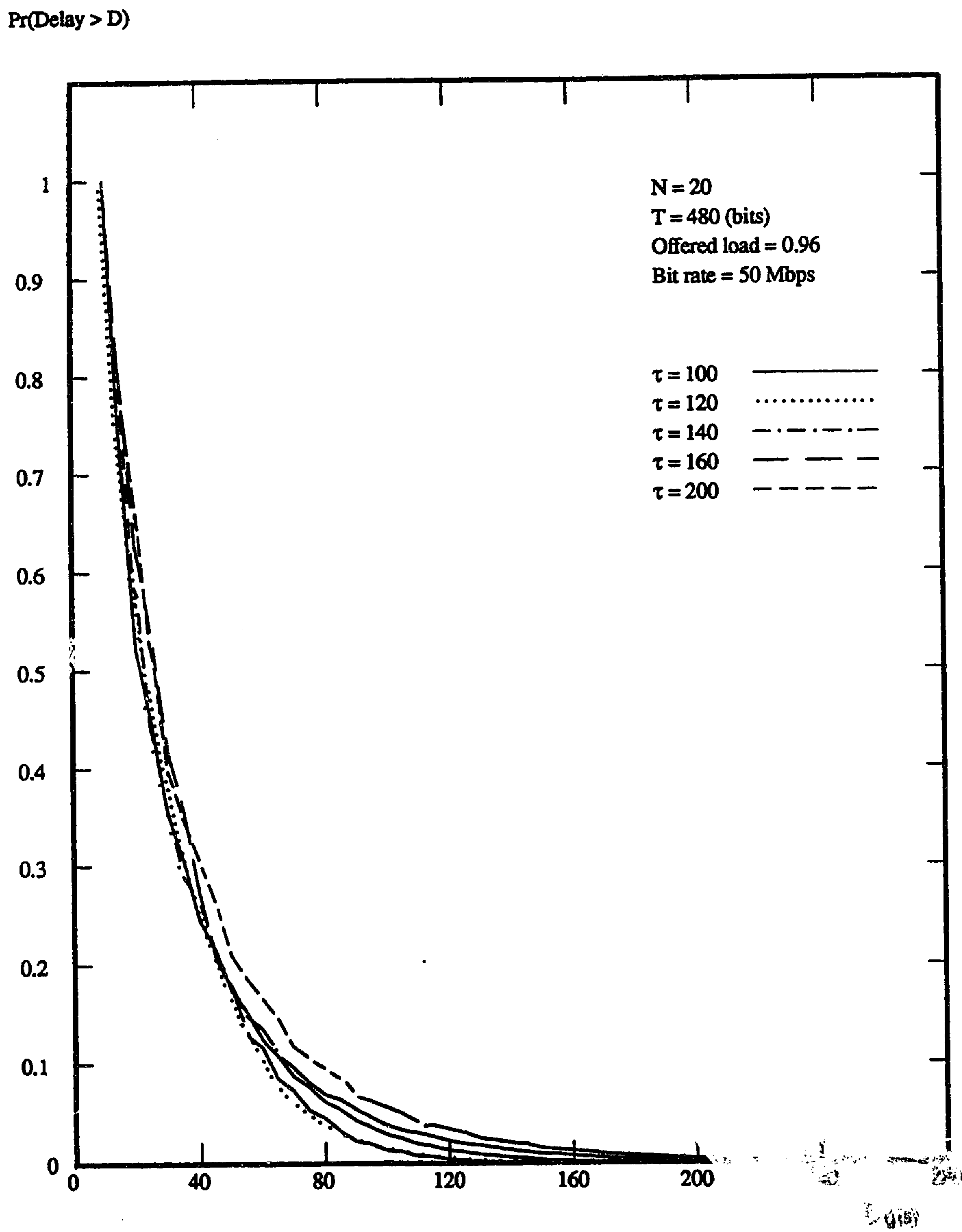


Figure 4.2: Survivor function of CASLANs at different values of τ

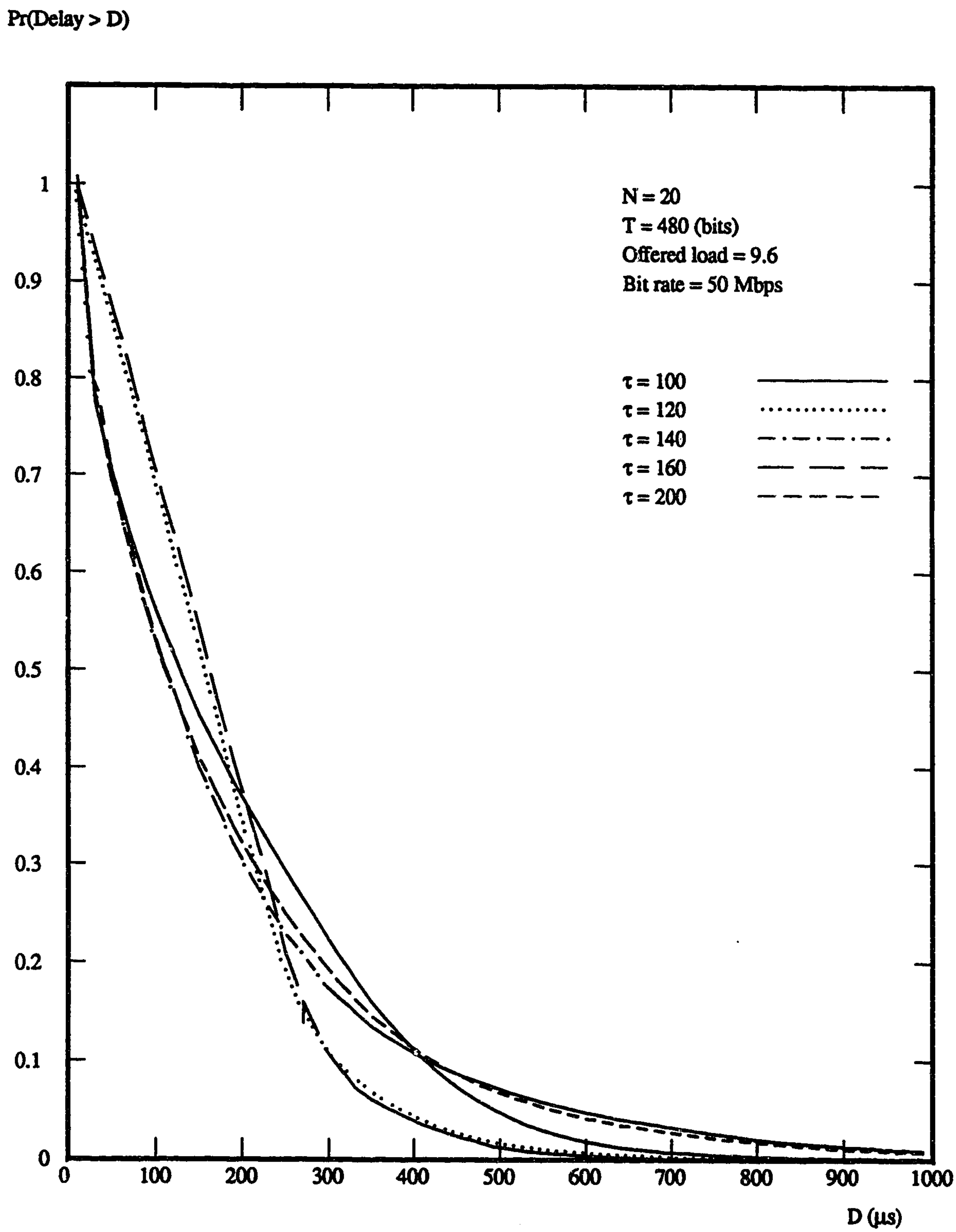


Figure 4.3: Survivor function of CASLANs at different values of τ

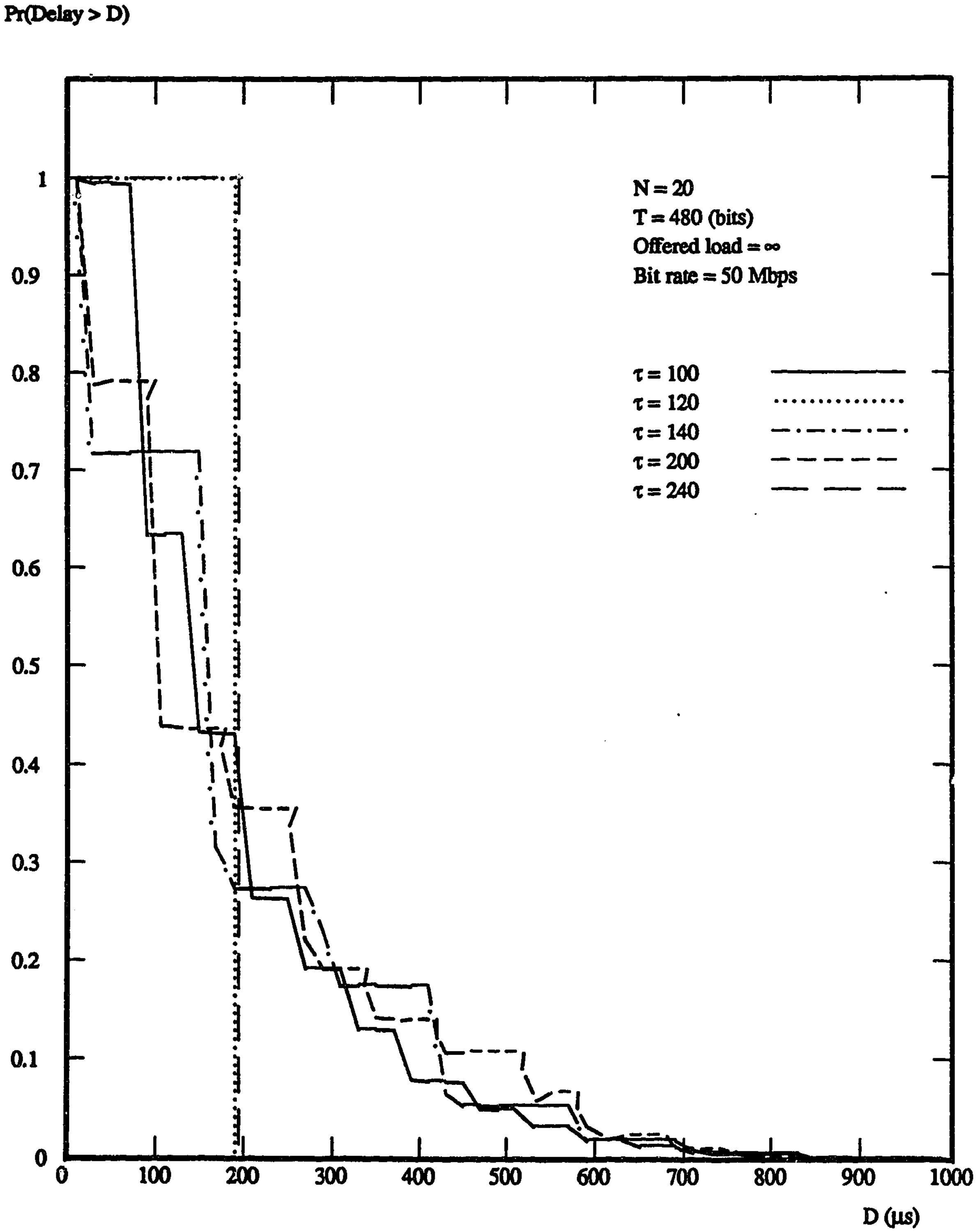


Figure 4.4: Survivor function of CASLANs at different values of τ

in packet survival. For instance, at an offered load = 0.96, the probability of loss at τ=120 (bits) is always lower than that at τ=100 (bits), see Figure 4.2. This phenomenon, of superior performance of factor retry time values, was earlier reported for average performance values, see section 2.1.

(4) At heavy load, figure 4.3, another interesting phenomenon arises. Factor retry time values seem to have a different shape of the probability density function (pdf), for packet delay, than non-factor retry time values. Indeed, at factor retry time values the pdf shape is narrower. This shape is reflected in the packet survival function in that factor retry time values seem to be worse than non-factor retry time values at low deadline values. At higher deadline values, this is completely reversed. For instance, at τ=120 and 160 (bits), the probability of packet loss is higher than that at other retry time values for packet deadlines less than 230 μs, see Figure 4.3. On the other hand, at a deadline of 400 μs, the probability of loss at τ=160 (bits) is only 4% as opposed to 12% at τ=100 (bits).

(5) The above phenomenon is much more apparent in a congested CASLAN, where the pdf at retry time values that are factors of the packet length reduces to the impulse function at $D = NT + \tau$. (This was earlier demonstrated in section 2.3) In other words, the probability of loss at such retry time values is 1 for deadline values less than $NT + \tau$ and 0 otherwise, see Figure 4.4.

The variable Packet length case is much more straight forward and the higher the retry time and/or the offered load the lower the packet survival rate. Performance values are very close to those in the fixed packet length case and are , hence, not shown.

4.3.2 Hard Real-Time CASLANs

Recall that in section 2.3, it was shown that *Congested* CASLANs with retry time values that are factors of the packet length operate in a round robin fashion with bounded delay. The delay is given by

$$D = N \cdot T + \tau$$

In real-time terms, the above implies that in a congested CASLAN with a retry time that is a factor of the packet length, a packet laxity that is no less than $N \cdot T + \tau$ would result in no packet loss. This, however, only applies to congested CASLANs where, as was shown, the packet transmission order is maintained. A more general hard real-time performance model of CASLANs is presented here. In this model, packet laxities are exponentially distributed. Packets that exceed their laxities are removed from the system. Although exponentially distributed laxities are not representative of most applications, the inclusion of any other distribution of laxities makes the analysis intractable. Therefore, for those applications that use a different distribution of laxity, the exponential distribution can be at least regarded as an approximation.

The model is an extension of the model described in section 3.2. That is, a tagged user is modeled exactly and the behavior of the rest of the users is approximated to follow the steady state behavior of that tagged user in an iterative approach. This model is described in the following section.

4.3.2.1. The Model

In general, modeling a hard real-time system is not a simple task. This is mainly due to the fact that such models must keep track of packet delays. Packets whose delays exceed the application imposed deadline are discarded. Hence, the evolution of the system state may be dependent on too many involved factors. In most cases, the mathematical models are intractable. However, by the use of some simplifying assumptions, and in the case of some special systems, the complexity of the problem can be reduced significantly, or at least the analysis can be made tractable.

For the single server queue, the hard real-time problem has been studied by several researchers. For instance, the M/M/1 queue [75-77], the M/G/1 queue [78-80] and the Geom/G/1 queue [81].

In [77] and [78], the probability that the unfinished work in the queue (W) exceeds a certain value (L) was computed. Now, by noting that the unfinished work in the queue corresponds to the waiting time an arriving packet would experience under FCFS scheduling, any packet facing an unfinished work greater than its laxity (L), on its arrival, would not join the queue. Kurose et al [73] implemented such a scheme on a FCFS window protocol and the results of the model were very accurate.

Rubin and Ouaily [80, 81] derived two hard real-time models for the M/G/1 and the Geom/G/1 queues. In the first model, a packet is always admitted to the queue, and if it exceeds its laxity, it is removed from the queue. In the second model, a packet is not admitted to the queue unless it is determined that it will be eventually transmitted. As expected, it was shown that the second approach yields better performance results.

Neither the latter model above nor the unfinished work model used in [73] can be used in modeling CASLANs. This is because:

(1) In CASLANs, as in many other distributed LAN protocols, stations have no knowledge of other packets in the system.

(2) Because of the retry nature of CASLANs, packets are served in random order. Therefore, a FCFS queue model is no longer applicable.

The model presented here is based on following a tagged user (in an exact analytical approach), while the treatment of the rest of the users is through an approach that involves some simplifying approximations. All packets are admitted to the system regardless of whether they are to be successful or not. Packets exit the system if they are successfully transmitted or if their waiting times exceed their corresponding laxities.

Assumptions used:

(1) Except for the tagged user, the retry instants for packets of all nodes are independent of each other and of the packet length. They are identically distributed and are taken from a general, unknown distribution that is later taken from the steady state behavior of the tagged user.

(2) All nodes are equipped with single-packet occupancy buffers. That is, once a packet occupies the buffer, no new packets are generated at a node unless the packet currently in the buffer has been successfully transmitted.

(3) Identical and independent packet generation process at all nodes. The packet generation time is exponentially distributed with rate λ.

(4) Packet laxities are exponentially distributed with rate μ.

(5) Fixed and equal packet length (T) for all nodes.

(6) The round trip propagation delay between any node and the hub is the same for all nodes, and is equal to the retry time (τ).

(7) The packet transmission time is at least equal to the retry time. That is,

$$T \geq \tau$$

As in the model of chapter 3, the model used in this section is based on a discrete time approach. The retry time is divided into a number of slots, g, which is referred to as the grid size. An embedded Markovian chain model is used with the embedding points being at the start of hub acquisitions.

Let ζ, K, Δ and σ be as defined in section 3.2. We also define α as the probability of loss in a slot. α is given by

$$\alpha = 1 - e^{-\mu\zeta}$$

Because of the discrete nature of the model, it is assumed that packets that arrive at the hub during a slot are delayed until the beginning of the next slot. This implies that the packets that contend for the acquisition of the hub at the beginning of a slot are those that arrived at the hub during the preceding slot.

In this model, and as the model in chapter 3, the system is represented as a polling one. The hub chooses at random one of the following four groups (see Figure 4.5):

- Idle users. These are the users with no packets to transmit at the end of the hub acquisition. They may then generate a packet in an idle slot before the next hub acquisition. If they do, they may only acquire the hub at the beginning of the slot following the packet generation slot.
- First-time ready users. Those are the users which became ready during the last hub acquisition.
- Carry-over ready users. Those are the ready users that were ready before the start of the last hub acquisition.
- The tagged user, which may or may not be ready.

Only packets from i carry-over ready users are subject to loss. The remainder of the carry-over ready users are inspected twice during a transition. Once at the end of the hub acquisition (i') and the other at the end of the following overhead period (i''), see Figure 4.5. In the sequel, these would correspond to $P_{loss}(\cdot)$ and $P_{ready\_loss}(\cdot)$, respectively.

A three-dimensional embedded Markovian chain is used to represent the polling system, where the embedding points are at the start of hub acquisitions.

1. The first dimension is the state in which the tagged user is in. We consider three different states for the tagged user, namely: $\bar{R}$ (not ready), A (Acquiring the hub), and R (ready),

2. the second is the number of carry-over ready users, and

3. the third is the remainder of the hub acquisition time at the instant of arrival of the tagged users packet (Y), if the tagged user is ready. If the tagged user is not ready, Y is arbitrarily set to zero.

In such a representation, $P(new, j, y_l \mid old, i, y_{l-1})$ is the transition probability from state (old, i, y_{l-1}) to state (new, j, y_l), where *old* (*new*), i (j), and y_{l-1} (y_l) are as defined in section 3.2.

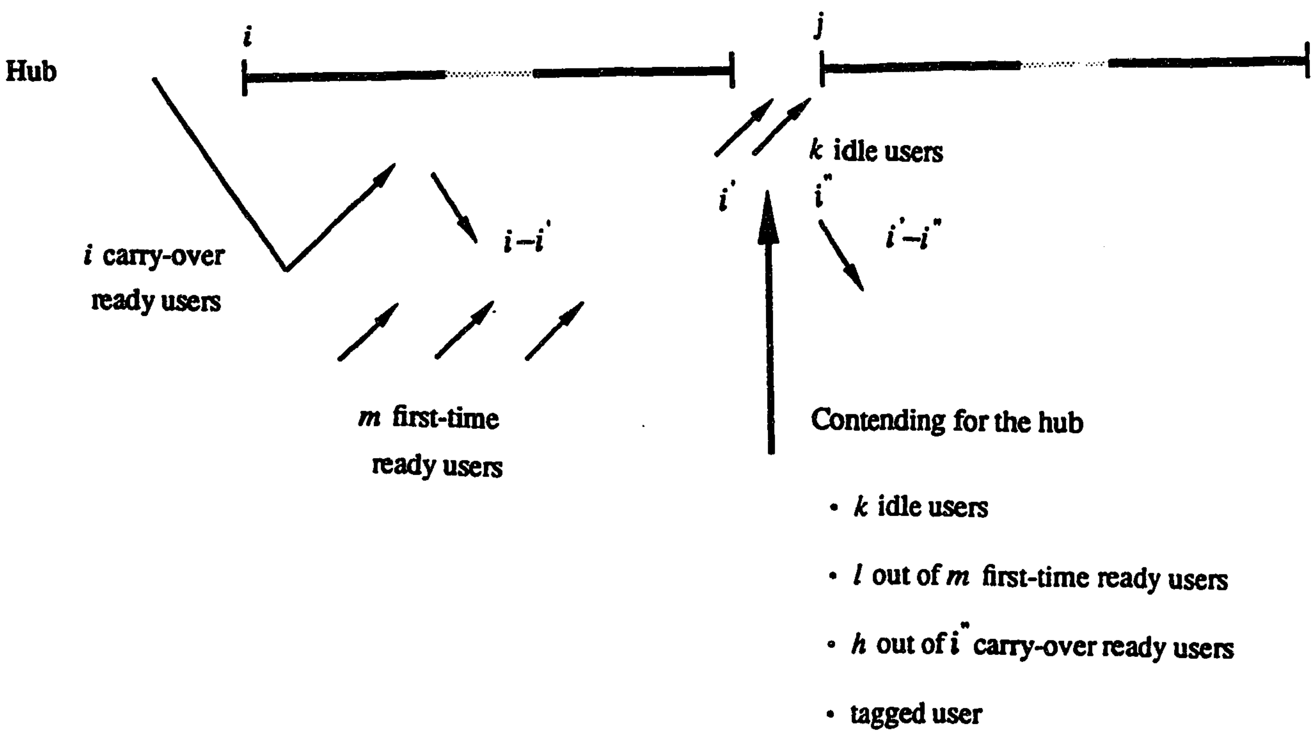


Figure 4.5 : Illustration of the different classes of users contending for the hub

To simplify the expressions of the transition probabilities, the following definitions are made. First, let $P_x(u)$, $P_\alpha(n,M,T)$, $P_L(t)$, $P_{\bar{L}}(t)$, $P_{idle}(k,M,t)$, $P_{ready}(h,M,t)$ and $P_{new}(l,M,t)$ be as defined as in section 3.2. The following additional definitions are made.

- $P_s(u) = 1-(1-\alpha)^u$

  This is the probability that a packet from a carry-over ready user is lost during time u.

- $$P_{loss}(n,M,T) = \begin{cases} \binom{M}{n} P_s(T)^{M-n} (1-P_s(T))^n & 0 \le n \le M \\ 0 & otherwise \end{cases}$$

  This is the probability that $M-n$ packets out of M carry-over ready nodes are lost during the transmission time, T.

- $$P_{ls}(t) = \sum_{n=1}^{t} (1-(1-\alpha)^{n-1}) \cdot P_{arr}(t) = \left[t - \frac{1-(1-\alpha)^t}{\alpha} \right] \cdot P_{arr}(t)$$

  This is the probability that a packet from a carry-over ready user is lost during an overhead period of length t without contending for the hub. $P_{arr}(t)$ is simply the probability that a packet from a carry-over ready user arrives at the hub exactly in slot t.

- $$P_{ready\_loss}(n,M,t) = \begin{cases} \binom{M}{n} P_{ls}(t)^{M-n} (1-P_{ls}(t))^n & 0 \le n \le M \\ 0 & otherwise \end{cases}$$

  This is the probability that $M-n$ packets out of M carry-over ready nodes are lost during the overhead period, t.

It can be noticed that, the above probabilities are all functions of the overhead period t. Since t is dependent on the number of ready users in the system, as well as the interarrival times of the tagged user, one can first obtain the joint transition probabilities $P^{(t)}(new, j\, y_{aft} \mid old, i\, y_{bef})$, then sum over all values of t. Depending on the state of the tagged user, 9 different cases exist. Expressions for transition probabilities of each of those cases are shown below.

Transition Probability Expressions

First, the following assumptions regarding packet loss are introduced:

(1) Other than the tagged user, only packets from carry-over ready users are subject to loss. This is a reasonable and justifiable assumption, since idle users do not have packets and first-time ready users' packets at the end of a transition would have arrived at the hub no more than $T+t$ slots earlier.

(2) In a transition, if a packet is lost during the hub acquisition, it is assumed to have been lost since the start of the transition.

(3) If the tagged user's packet is lost, the tagged user is not allowed to generate another packet before the start of the next transition.This assumption is made to reduce the complexity of computing the probabilities of packet loss. Without this assumption, computing the loss probability would have been a lengthy and very involved process, without significantly enhancing the accuracy of the model.

(4) (1) and (3) above imply that packets from the tagged user could only be lost if the tagged user went from ready to not ready.

I. Tagged user is initially not ready:

Li Case $\bar{R} \to \bar{R}$:

The tagged user is not ready and will not become ready during the transition.

$$P^{(t)}(\bar{R}, j, y_{aft} \mid \bar{R}, i, y_{bef}) = (1-P_x(t+T)) \cdot \sum_{i'=0}^{i} P_{loss}(i', i, T) \sum_{i''=0}^{i'} P_{ready\_loss}(i'', i', t)$$

$$\sum_{k=0}^{j-i''+1} \sum_{l=0}^{j-K-i''+1} \sum_{h=0}^{i''}$$

$$\left\{ P_{\bar{L}}(t) \cdot P_{tx}(N-i'-1, j-k-i''+1, T) \cdot P_{idle}(k, N-j+k-i'+i''-2, t) \cdot P_{ready}(h, i'', t) \cdot P_{new}(l, j-k-i''+1, t) \right.$$

$$\left. + P_L(t) \cdot P_{tx}(N-i'-1, j-k-i'', T) \cdot P_{idle}(k, N-j+k-i'+i''-1, t) \cdot P_{ready}(h, i'', t) \cdot P_{new}(l, j-k-i'', t) \right\} \quad (4.1)$$

The term outside the summation in equation (4.1) above is the probability that the tagged user does not generate a packet during the transition. The first two sums are over all values of lost carry-over ready packets at the end of the hub acquisition and the following overhead period, respectively. The term inside the curly bracket is the probability that k idle, h carry-over ready and l new users have their packets arrive at the hub exactly in slot t after the hub release. Note that only one of these users would acquire the hub. The rest, if any, will become carry-over ready users at the next embedding point.

I.ii Case $\bar{R} \to A$

The tagged user becomes ready after a transition. The transition probability is given by

$$P^{(t)}(A,j,y_{aft} \mid \bar{R},i,y_{bef}) = \sigma(1-\sigma)^{t-1} \cdot \sum_{i'=0}^{i} P_{loss}(i',i,T) \sum_{i''=0}^{i'} P_{ready\_loss}(i'',i',t)$$

$$\sum_{k=0}^{j-i''} \sum_{l=0}^{j-k-i''} \sum_{h=0}^{i''} \frac{1}{k+l+h+1} \Big\{ P_{\bar{L}}(t) \cdot P_{tx}(N-i'-1,j-k-i'',T) \cdot P_{idle}(k,N-j+k-i'+i''-1,t) \cdot$$

$$P_{ready}(h,i'',t) \cdot P_{new}(l,j-k-i'',t) + P_L(t) \cdot P_{tx}(N-i'-1,j-k-1,T) \cdot P_{idle}(k,N-j+k-i'+i''-1,t) \cdot$$

$$P_{ready}(h,i'',t) \cdot P_{new}(l,j-k-i''-1,t) \Big\} \tag{4.2}$$

The term outside the summation in equation (4.2) above is the probability that the tagged user packet arrives exactly in slot t. The inside summations are over all possible combinations of having k idle, h carry-over ready and l new users schedule their packet arrivals at the hub, exactly t slots after the hub release. Again, the probability accounts for both cases in which the user that last released the hub is involved or not.

I.iii Case $\bar{R} \to R$

The tagged user becomes ready. Let Z be the first interarrival time of the packet from the tagged user. Now $X = T - Z$ (refer to Figure 3.4). X can be in one of two ranges, and the transition probability depends on which range X is in.

$1 \le X \le T$:

This is the normal case where the tagged user's packet arrival finds a busy hub, see Figure 3.4a. The transition probability is given by

$$P^{(X,t)}(R,j,y_{aft} \mid \bar{R},i,y_{bef}) = \sigma(1-\sigma)^{t+T-X-1} \cdot \sum_{i'=0}^{i} P_{loss}(i',i,T) \sum_{i''=0}^{i'} P_{ready\_loss}(i'',i',t)$$

$$\sum_{k=0}^{j-i''+1}\sum_{l=0}^{j-k-i''+1}\sum_{h=0}^{i''}\Big\{P_{\bar{L}}(t)\cdot P_{tx}(N-i'-1,j-k-i''+1,T)\cdot P_{idle}(k,N-j+k-i'+i''-2,t)\cdot$$

$$P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i''+1,t)+P_L(t)\cdot P_{tx}(N-i'-1,j-k-i'',T)\cdot P_{idle}(k,N-j+k-i'+i''-1,t)\cdot$$

$$P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i'',t)\Big\} \tag{4.3}$$

The term outside the summation is the probability that the tagged user becomes ready at exactly slot $t+T-X$. The terms inside the curly brackets are the same as those in (4.1).

<u>$X=T+1$:</u>

This case is different from the above because the tagged user will contend for the hub, see Figure 3.4b. The transition probability is given by

$$P^{(X,t)}(R,j,y_{aft}\,|\,\bar{R},i,y_{bef})=\sigma(1-\sigma)^{t-1}\cdot\sum_{i'=0}^{i}P_{loss}(i',i,T)\sum_{i''=0}^{i'}P_{ready\_loss}(i'',i',t)$$

$$\sum_{k=0}^{j-i''+1}\sum_{l=0}^{j-i''-k+1}\sum_{h=0}^{i''}\frac{k+l+h}{k+l+h+1}\cdot\Big\{P_{\bar{L}}(t)\cdot P_{tx}(N-i'-1,j-k-i''+1,T)\cdot P_{idle}(k,N-j+k-i'+i''-2,t)\cdot$$

$$P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i''+1,t)+P_L(t)\cdot P_{tx}(N-i'-1,j-k-i'',T)\cdot P_{idle}(k,N-j+k-i'+i''-1,t)\cdot$$

$$P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i'',t)\Big\} \tag{4.4}$$

The difference between equation (4.3) and (4.4) is the quotient outside the curly brackets in equation (4.4), since this accounts for the total number of users contending for the hub, which, in this case, includes the tagged user.

Now, recalling that $X=n\tau+y_{aft}$ and unconditionning on X, then

$$P^{(t)}(R,j,y_{aft}\mid \bar{R},y_{bef}) = \begin{cases} \sum_{n=0}^{K} P^{(n\tau+y_{aft},t)}(R,j,y_{bef}\mid \bar{R},i,y_{bef}) & 0\le y_{aft}<\Delta \\ \sum_{n=0}^{K-1} P^{(n\tau+y_{aft},t)}(R,j,y_{aft}\mid \bar{R},i,y_{bef}) & \Delta\le y_{aft}<\tau \end{cases} \tag{4.5}$$

II. Tagged user is initially acquiring the hub

II.i Case $A\to\bar{R}$:

$$P^{(t)}(\bar{R},j,y_{aft}\mid A,i,y_{bef}) = (1-P_x(t+T-\tau))\sum_{i'=0}^{i} P_{loss}(i',i,T)\sum_{i''=0}^{i'} P_{ready\_loss}(i'',i',t)$$

$$\sum_{k=0}^{j-i''+1}\sum_{l=0}^{j-k-i''+1}\sum_{h=0}^{i}$$

$$P_{tx}(N-i',j-k-i''+1,T)\cdot P_{idle}(k,N-j+k-i'+i''-1)\cdot P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i''+1,t) \tag{4.6}$$

The term outside the summation in equation (4.6) represents the probability of no packet arrival from the tagged user. The term inside the summation is the probability that k, h and l packets from the idle, carry-over ready and first-time ready users, respectively, arrive at exactly slot t.

II.ii Case $A\to A$

For the tagged user to reacquire the hub on the very next hub acquisition, then t must be greater than τ. Thus, there should be no remaining ready users to contend for the hub. That is, all ready packets must be lost during the transition. The transition probability is therefore given by

$$P^{(t)}(A,j,y_{aft} \mid A,i,y_{bef}) = \sigma(1-\sigma)^{t-\tau-1} \cdot P_{loss}(0,i,T+\tau) \cdot P_{idle}(j,N,t) \cdot P_{tx}(N,0,T) \cdot \frac{1}{1+j} \tag{4.7}$$

In (4.7) above, the first term is the probability of a packet arrival from the tagged user at exactly slot t, the second term is the probability that all current carry-over ready users are to be lost during the transition, the third term is the probability of the arrival of j packets out of N idle users, while the fourth is the probability of no new arrivals during the tagged user first transmission. The last term is the probability that the hub randomly chooses the packet from the tagged user over the other j packets that arrived in slot t.

II.iii Case $A \rightarrow R$

This case is similar to the $\bar{R} \rightarrow R$ case, except for two differences. First, the tagged user must wait for a time equal to τ before being able to generate its next packet. This is because the tagged user is the last to release the hub. Second, one must distinguish between the cases for which $t \le \tau$ and $t > \tau$.

- $1 \le t \le \tau$

It is not hard to see, in this case, that $1 \le X \le T+t-\tau$. The transition probability is given by

$$P^{(X,t)}(R,j,y_{aft} \mid A,i,y_{bef}) = \sigma(1-\sigma)^{t+T-\tau-X} \cdot \sum_{i'=0}^{i} P_{loss}(i',i,T) \sum_{i''=0}^{i'} P_{ready\_loss}(i'',i',t)$$

$$\sum_{k=0}^{j-i'+1} \sum_{l=0}^{j-k-i'+1} \sum_{h=0}^{i''} P_{tx}(N-i',j-k-i''+1,T) \cdot P_{idle}(k,N-j+k-i'+i''-1)$$

$$\cdot P_{ready}(h,i'',t) \cdot P_{new}(l,j-k-i''+1,t) \tag{4.8}$$

- $t > \tau$

Here again a distinction between the case where $1 \le X \le T$ and the case where $X = T+1$ is made. The latter represents the case where the tagged user contends for the hub. Note that because of the limits on X, this case was non-existent with $t \le \tau$.

<u>$1 \le X \le T$</u>

The transition probability in this case is the same as (4.8) except for the limits on X.

<u>$X = T+1$</u>

The tagged user contends for the hub. The transition probability is given by

$$P^{(X,t)}(R,j,y_{aft} \mid A,0,y_{bef}) = \sigma(1-\sigma)^{t-\tau-1} \cdot \sum_{i'=0}^{i} P_{loss}(i',i,T) \sum_{i''=0}^{i'} P_{ready\_loss}(i'',i',t)$$

$$\sum_{k=0}^{j-i''+1} \sum_{l=0}^{j-k-i''+1} \sum_{h=0}^{i''} \frac{k+l+h}{1+k+l+h} \cdot P_{tx}(N-i',j-k-i''+1,T) \cdot P_{idle}(k,N-j+k-i'+i''-1) \cdot$$

$$P_{ready}(h,i'',t) \cdot P_{new}(l,j-k-i''+1,t) \tag{4.9}$$

As in the $\bar{R} \to R$ transition

$$P^{(t)}(R,j,y_{aft} \mid A,i,y_{bef}) = \begin{cases} \sum_{n=0}^{K} P^{(n\tau+y_{aft},t)}(R,j,y_{aft} \mid A,i,y_{bef}) & 0 \le y_{aft} < \Delta \\ \sum_{n=0}^{K-1} P^{(n\tau+y_{aft},t)}(R,j,yaft \mid A,i,y_{bef}) & \Delta \le y_{aft} < \tau \end{cases} \tag{4.10}$$

III. The tagged user is initially ready:

<u>III.i Case $R \to \bar{R}$</u>

As far as the hard real-time system is concerned, this case is one of the most important ones, since for the tagged user to become idle ($\bar{R}$) after being ready (R), its packet must be lost. If it is not lost, the tagged user must go through an $R \to A$, then an $A \to \bar{R}$ transitions.

A distinction between two different cases is made:

• Packet lost after $\tau-y_{bef}$:

In this case, $t\le\tau-y_{bef}$, see Figure 4.6. The transition probability is given by (4.11) below.

$$P^{(t)}(\bar{R},j,y_{aft}\mid R,i,y_{bef})=(1-\alpha)^{\tau-y_{bef}}\cdot(1-(1-\alpha))^{T+t-\tau+y_{bef}}\sum_{i'=0}^{i}P_{loss}(i',i,T)$$

$$\sum_{i''=0}^{i'}P_{ready\_loss}(i'',i',t)\sum_{k=0}^{j-i''+1}\sum_{l=0}^{j-K-i''+1}\sum_{h=0}^{i''}$$

$$\Big\{P_{\bar{L}}(t)\cdot P_{tx}(N-i'-1,j-k-i''+1,T)\cdot P_{idle}(k,N-j+k-i'+i''-2,t)\cdot P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i''+1,t)+$$

$$P_L(t)\cdot P_{tx}(N-i'-1,j-k-i'',T)\cdot P_{idle}(k,N-j+k-i'+i''-1,t)\cdot P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i'',t)\Big\} \quad (4.11)$$

The first term above is the probability that the tagged user's packet is lost after time $\tau-y_{bef}$. The remainder of the expression is the same as equation (4.1).

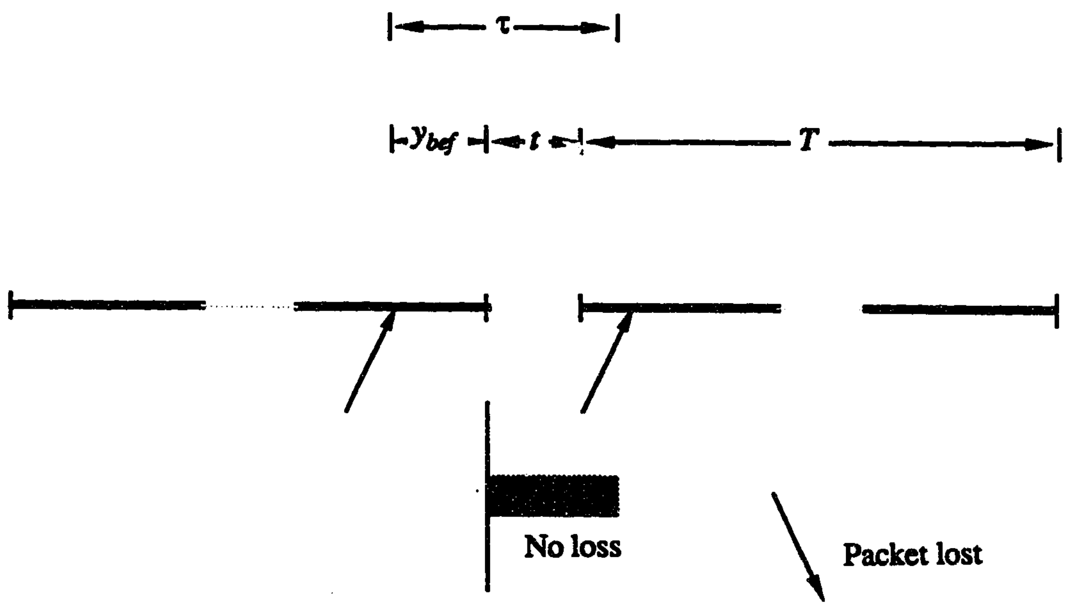
• Packet lost during $\tau-y_{bef}$:

This case is similar to the above case except for the first term. This term is replaced by $1-(1-\alpha)^{\tau-y_{bef}}$. Also, there is no limit on the overhead period.

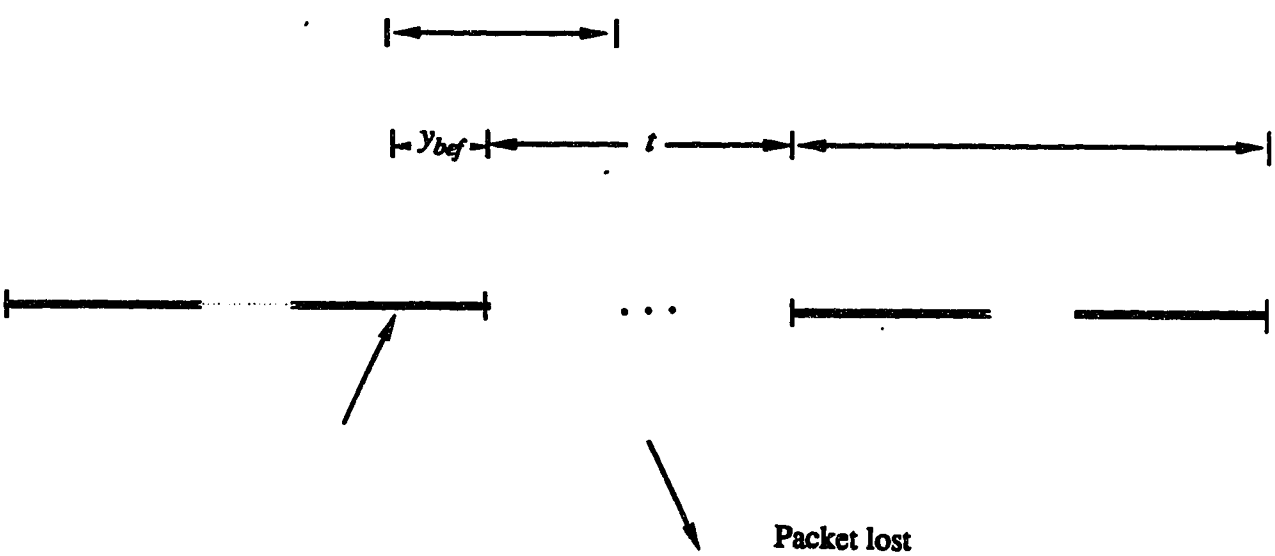
<u>III.ii Case $R\to A$</u>

Since the model keeps track of Y values of the tagged user, the value of the overhead period t is deterministic and is given by $t=\tau-y_{bef}$.

$$P(A,j,y_{aft}\mid R,i,y_{bef})=(1-\alpha)^{\tau-y_{bef}}\cdot\sum_{i'=0}^{i}P_{loss}(i',i,T)\sum_{i''=0}^{i'}P_{ready\_loss}(i'',i',t)$$



(a) Packet lost after arriving during the hub acqisition



(b) Packet lost during the overhead period

Figure 4.6 : An $R \rightarrow \bar{R}$ transition

$$\sum_{k=0}^{j-i''}\sum_{l=0}^{j-k-i''}\sum_{h=0}^{i''}\frac{1}{k+l+h+1}$$

$$\Big\{P_{\bar{L}}(t)\cdot P_{tx}(N-i'-1,j-k-i'',T)\cdot P_{idle}(k,N-j+k-i'+i''-1,t)\cdot P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i'',t)+$$

$$P_L(t)\cdot P_{tx}(N-i'-1,j-k-1,T)\cdot P_{idle}(k,N-j+k-i'+i''-1,t)\cdot P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i''-1,t)\Big\} \quad (4.12)$$

This is the same as the $\bar{R}\rightarrow A$ transition, equation (4.2), except for the term outside the summation. This term in (4.2) is replaced by the probability of no loss during $\tau-y_{bef}$ in equation (4.12).

III.iii Case $R\rightarrow R$

For a transition from y_{bef} to y_{aft}, the overhead period t is deterministic, see Figure 3.7, and is given by

$$t=\tau-y_{bef}+X-T$$

where

$$X=\begin{cases}K\tau+y_{aft} & 0\le y_{aft}\le\Delta\\ (K-1)\tau+y_{aft} & \Delta\le y_{aft}<\tau\end{cases} \quad (4.13)$$

Therefore,

$$P(R,j,y_{aft}\,|\,R,i,y_{bef})=(1-\alpha)^{T+t}\cdot\sum_{i'=0}^{i}P_{loss}(i',i,T)\sum_{i''=0}^{i'}P_{ready\_loss}(i'',i',t)\sum_{k=0}^{j-i''+1}\sum_{l=0}^{j-k-i''+1}\sum_{h=0}^{i''}$$

$$\Big\{P_{\bar{L}}(t)\cdot P_{tx}(N-i'-1,j-k-i''+1,T)\cdot P_{idle}(k,N-j+k-i'+i''-2,t)\cdot P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i''+1,t)+$$

$$P_L(t)\cdot P_{tx}(N-i'-1,j-k-i'',T)\cdot P_{idle}(k,N-j+k-i'+i''-1,t)\cdot P_{ready}(h,i'',t)\cdot P_{new}(l,j-k-i'',t)\Bigg\} \quad (4.14)$$

The first term above is the probability that the tagged user's packet is not lost during time $t+T$. The remainder of the expression is the same as equation (4.1).

To solve for the steady state probabilities, a transformation from the three-dimensional state space to a one-dimensional state space is required. Such a transformation was described in section 3.2. One can then find the steady state probabilities by solving the set of equations in (4.15) below.

$$\vec{\pi} = \vec{\pi}\cdot\mathbf{P} \quad \text{and} \quad \vec{\pi}\cdot\vec{1} = 1 \qquad (4.15)$$

where,

P is the transition probability matrix.

$\vec{\pi}$ is a row vector containing the steady state probabilities.

$\vec{1}$ is a column vector of 1 elements.

4.3.2.2. Model Analysis

The most important performance measure considered is the probability of packet loss. The nature of the model makes such a requirement readily available once the transition probabilities and the steady state probabilities have bean obtained.

Since only packets from ready nodes can be lost and since the tagged user is not allowed to generate a new packet immediately after it loses one, the probability of loss can be expressed as the fraction of packets lost out of all the packets generated. Notice that, according to the model presented in section II, the tagged user loses a packet if a transition from R to $\bar{R}$ takes place. Also, the packets generated by the tagged user are either transmitted, therefore resulting in a state of A (acquired hub by the tagged user) at a transition, or are lost, hence resulting in the transition from R to $\bar{R}$. The probability of loss can therefore be expressed as:

$$Pr_{loss} = \frac{Pr(R \text{ followed by } \bar{R})}{Pr(R \text{ followed by } \bar{R}) + Pr(\text{acquired hub})}$$

$$= \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{N} \sum_{y_{bef}=0}^{\tau-1} P(\bar{R}, j, 0 \mid R, i, y_{bef}) \cdot \pi_{(R,i,y_{bef})}}{\sum_{i=0}^{N-1} \sum_{j=0}^{N} \sum_{y_{bef}=0}^{\tau-1} P(\bar{R}, j, 0 \mid R, i, y_{bef}) \cdot \pi_{(R,i,y_{bef})} + P(A)} \tag{4.16}$$

Notice that under steady state operation, and for a symmetric system with $N+1$ users,

$$P(A) = \frac{1}{N+1}$$

To enhance the model, the iterative method described in section 3.3.2 is employed. Recall that the main approximation in the model described in section 4.3.2.1 is the treatment of carry-over ready users. A uniform distribution was used to approximate their retry times. Here, a better approximation is presented for the distribution of carry-over ready users.

Define $P_{tag}(t \mid R)$ as the probability that the tagged user arrives at the hub exactly in slot t after the hub release given that the tagged user was ready. $P_{tag}(t \mid R)$ is given by

$$P_{tag}(t \mid R) = \frac{1}{P_{tag}(R)} \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{N} \sum_{y_{bef}=0}^{\tau-1} [P(A, j, 0 \mid R, i, y_{bef}) + P(R, j, y_{aft} \mid R, i, y_{bef})$$

$$+P(\bar{R},j,0|R,i,y_{bef})\cdot(1-\alpha)^{t}]\cdot\pi_{(R,i,y_{bef})} \quad (4.17)$$

where,

$$y_{bef}=\tau-t$$

and $P_{tag}(R)$ is the probability that the tagged use is ready and is given by

$$P_{tag}(R)=\sum_{i=0}^{N-1}\sum_{y_s=0}^{\tau-1}\pi_{(R,i,y_s)} \quad (4.18)$$

Since the tagged user is modeled exactly, a better approximation in modeling carry-over ready users is to use $P_{tag}(t|R)$ rather than the uniform distribution used earlier. The model is therefore modified to employ an iterative method in which $P_{tag}(t|R)$ values are collected and fed back to the model to represent the distribution of carry-over ready users.

$P_{ready}(h,M,t)$ is now redefined as

$$P_{ready}(h,M,t)=\begin{cases}\binom{M}{h}P_{tag}(t|R)^{h}\,(1-PF_{tag}(t|R))^{M-h} & 0\le h\le M \;\&\; t<\tau\\ P_{tag}(t|R)^{M} & h=M \;\&\; t=\tau\\ 0 & otherwise\end{cases}$$

where, $P_{tag}(t|R)$ is as defined above and $PF_{tag}(t|R)$ is the probability that a packet from a carry-over ready user arrives at the hub during time t from the hub release, and is given by

$$PF_{tag}(t|R)=\sum_{i=1}^{t}P_{tag}(i|R)$$

As should be obvious, and as mentioned in section 3.3.2, the larger the number of iterations and/or the grid size used, the more accurate the results. It was shown, however, in section 3.3.2 that a grid size of 20 and only 4 iterations are sufficient to obtain reasonably accurate results. Therefore, in obtaining the performance results,such values will be used.

4.3.2.3. Numerical Results:

The effect of the offered load, the number of nodes, the packet retry times and the mean packet laxities on the probability of loss is studied.

Values of the mean packet laxity are chosen such that to allow at least N packet transmission, on average, while the packet is waiting to access the channel. For example, if $N=20$ and $T=480$, then at least 9,600 bits plus the round trip propagation delay are required for N transmissions (about 200μs). In the following tests two values of mean packet laxity are used. These values are chosen to roughly represent NT and $2NT$.[19]

In Figure 4.7, the probability of packet loss versus the retry time is shown, (with T=480 bits, N=20) under light to medium loads at different values of mean laxities of 200 μs (10,000 bits) and 400 μs (20,000 bits). Results of the analytical model together with results from a simulation model are presented. The following observations can be easily made:

(1) CASLANs possess excellent survival probability at light to medium load. The highest reported probability of loss at offered load = 0.48 is less than 6%. At an offered load=0.096, which corresponds to light load, the loss probability is always

19. Even though these values may seem high, for reasonably large systems (e.g. 50 nodes), the packet laxities would still be in the 100's of μs range.

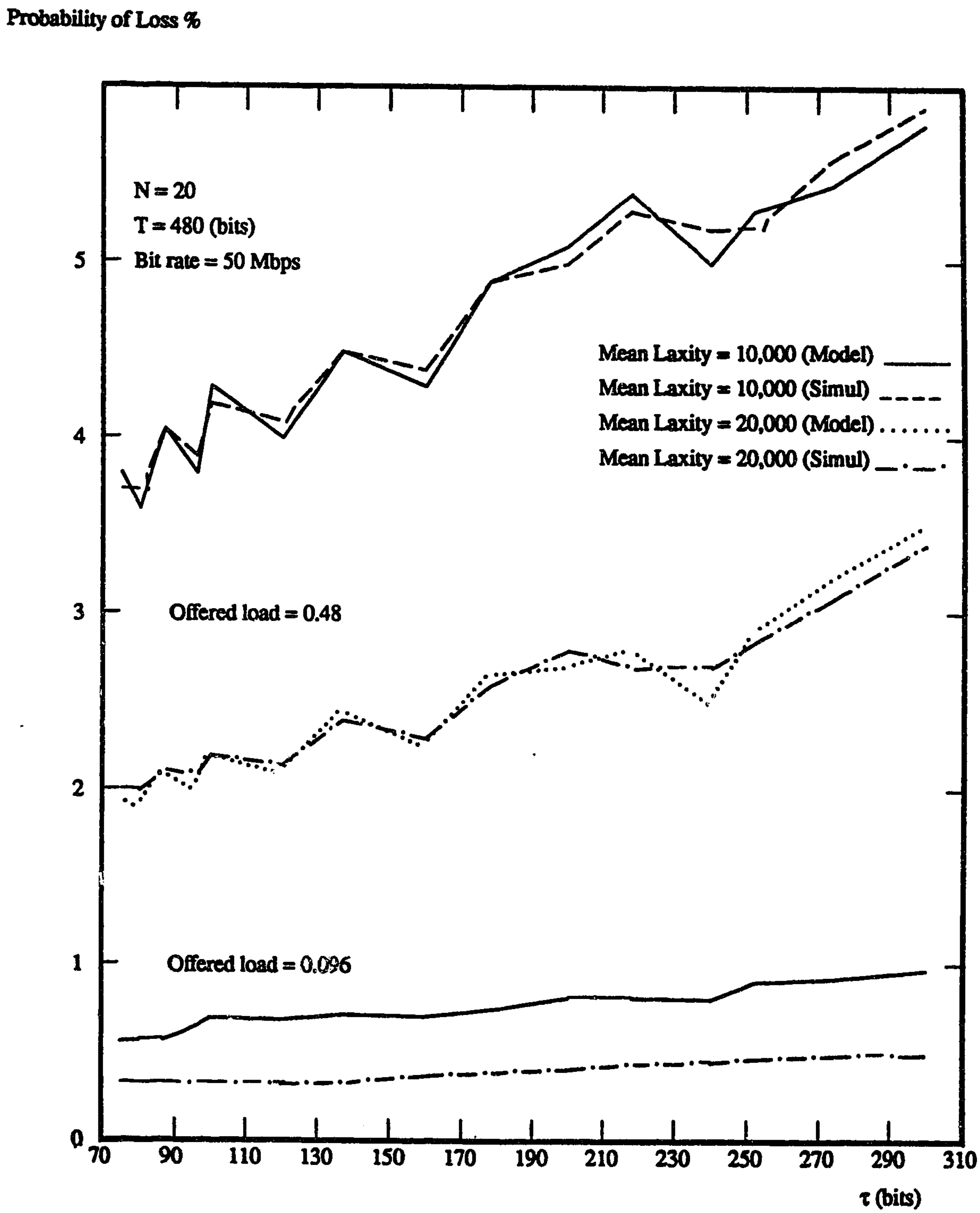


Figure 4.7: Probability of loss at different values of τ in a 20 node system

below 1%.

(2) In most cases, the probability of loss increases with increasing the retry time and/or the offered load, while it decreases with increasing the mean packet laxity. For example, in the case of applications with greater laxities, e. g., on the order of a millisecond or more, the probability of loss becomes much less.

(3) Results of the model are usually very close to simulation results. At an offered load of .096, the two coincide, and are indistinguishable.

In Figures 4.8 and 4.9, probability of packet loss versus the retry time are shown (with T=480 bits and high to heavy offered loads) at N = 10 and 20 nodes respectively. Again, results from the analytical and simulation models are presented. We make the following observations:

(1) CASLANs performance under real time constraints is reasonable. The probability of loss, at an offered load of 0.96 (almost unity) and a mean laxity of 10,000 bits, does not exceed 10%. Increasing the offered load by an order of magnitude, i.e. to 9.6, more than half the packets generated will survive.

(2) The results show that retry time values that are factors of the packet length seem to possess local superior performance, in the sense that a relatively lower loss probability is observed at such retry time values. This superiority, however, is insignificant. It should be noted that the model is capable of capturing such an effect.

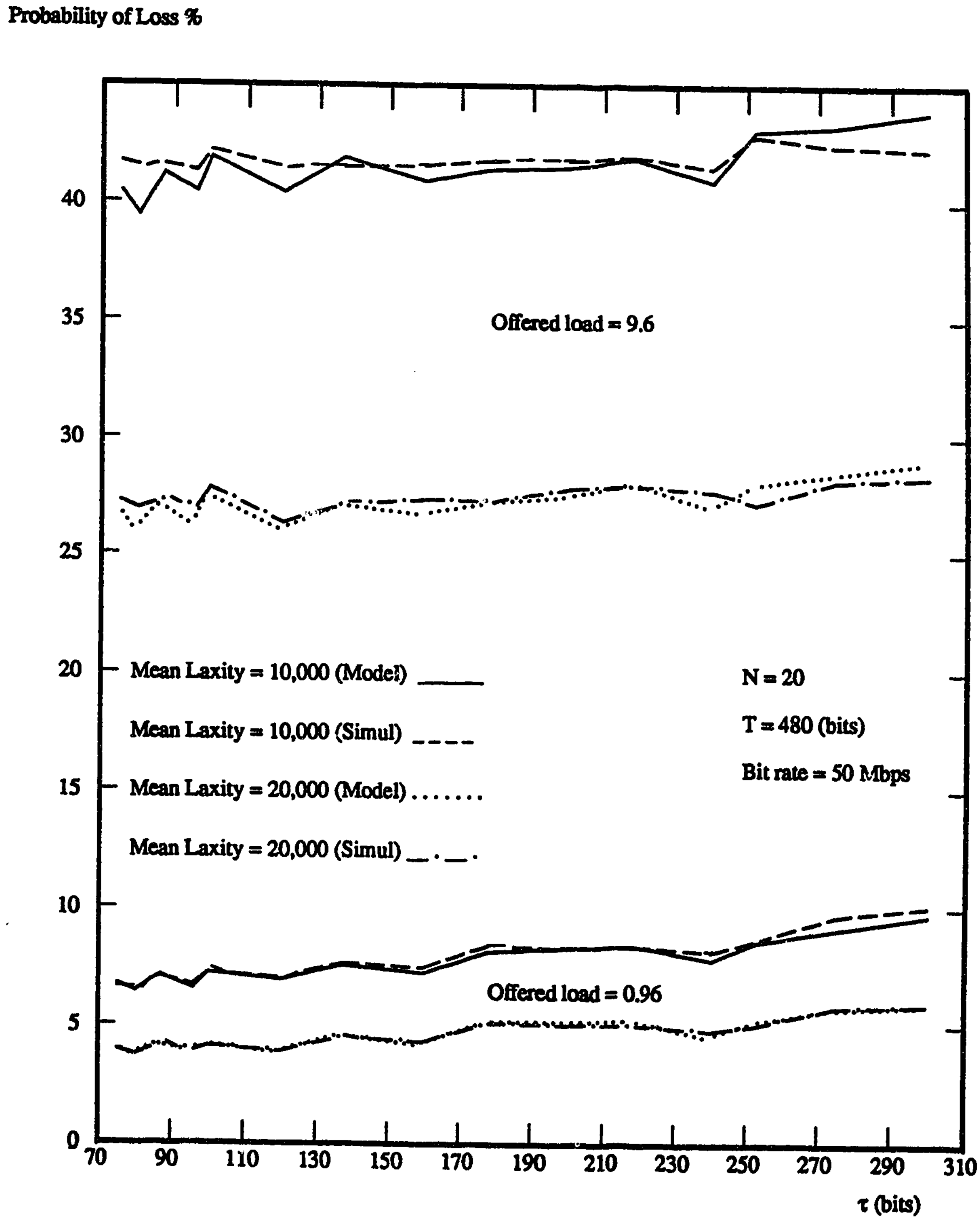


Figure 4.8: Probability of loss at different values of τ in a 20 node system

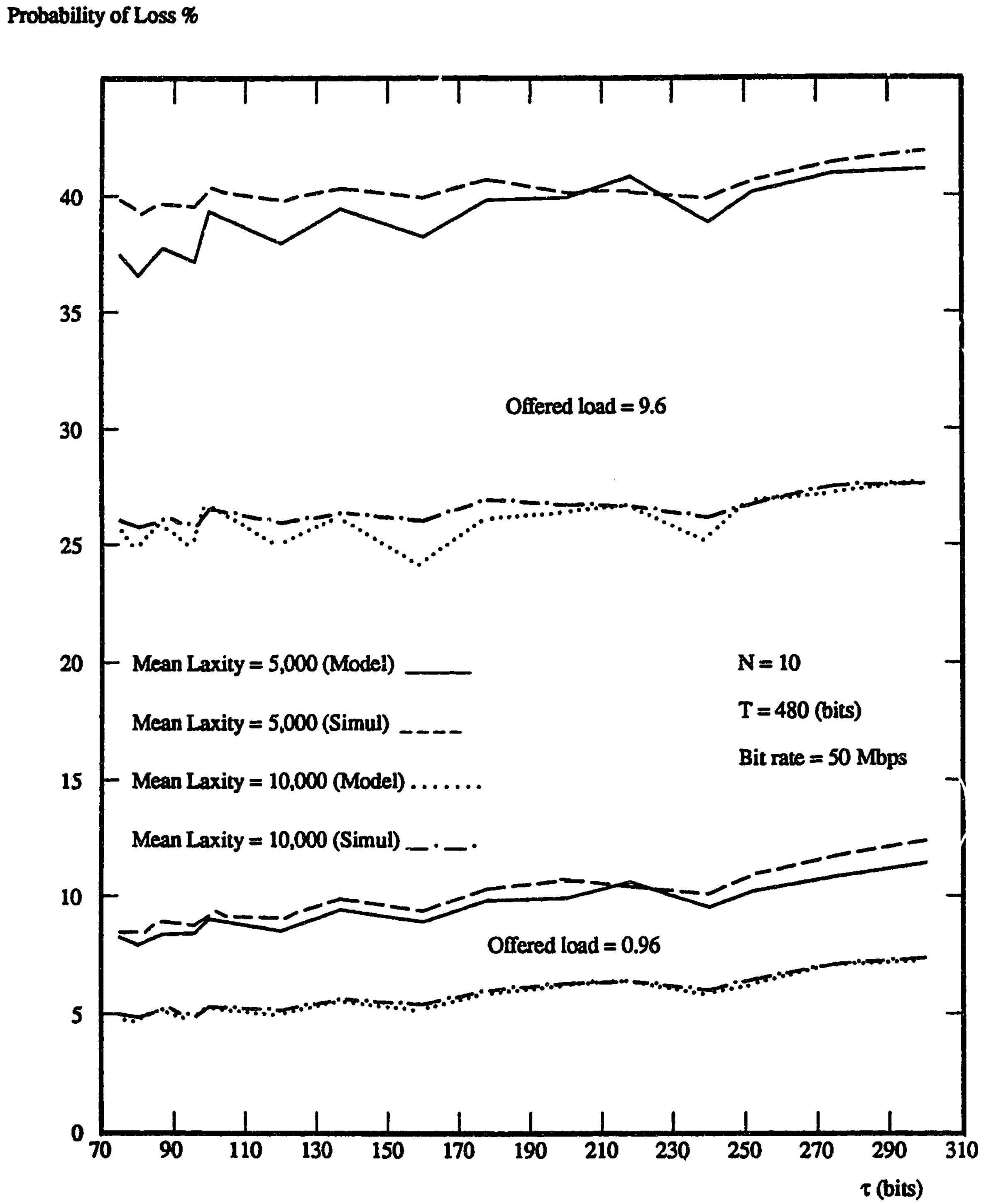


Figure 4.9: Probability of packet loss at different values of τ in a 10 node system

(3) The accuracy of the model is very reasonable. The maximum deviation from simulation results is less than 6%.

4.4. Summary

In this chapter, the performance of CASLANs in a real-time environment was studied. Both soft and hard real-time CASLANs were considered.

For soft real time CASLANs, a simulation model was devised from which the packet survival function was obtained. It was shown that CASLAN perform very well under soft real-time constraints. Also, it was shown that, at high to heavy load, factor retry time values have a higher survival rate than non-factor retry time values as deadline values increase.

For hard real-time CASLANs, a mathematical model analyzing the performance of such CASLANs was introduced. Results from this model were presented and compared to results from a simulation model.

The model is based on following a tagged user in a near-exact analysis and approximating the behavior of the rest of the users by substituting the steady state behavior of the tagged user. Packet laxities are sampled from an exponential distribution. Packets can exit the system either by acquiring the hub (successful transmission) or if they are lost.

The numerical results show that:

(1) In general, the accuracy of the performance model is very good since at extremely heavy load the error does not exceed 6%. Therefore, the model can be

used to accurately predict the performance of CASLANs in a hard real-time environment.

(2) Under light to medium offered loads, the performance of CASLANs under real-time constraints is excellent. Since most real-time systems are moderately loaded, it is projected that CASLANs are very good candidates for hard real-time applications.

(3) Even under extreme heavy load conditions, the survival rate of CASLANs operating with a mean laxity of 200 μs was above 50%. With a higher laxity, e.g., a millisecond, the probabilities of loss are greatly reduced and may even vanish.

(4) Retry time values that are factors of the packet length yield superior performance results.

Chapter 5

Prioritized CASLANs

Thus far in this thesis, only symmetric CASLANs were considered, i.e., CASLANs where all nodes have the same retry time and mean packet length. In this chapter, the performance of prioritized CASLANs is studied and analyzed. Emphasis is made on the effect of the retry time on priority implementation. Both data only and integrated voice and data applications are considered. It is shown that controlling retry time values might be sufficient to effectively implement priorities in CASLANs. During the course of studying prioritized CASLANs, a performance model is devised in section 5.4. The model is shown to accurately predict the performance of moderately loaded CASLANs.

The chapter begins by reviewing some prioritized access schemes that have been devised for use in random access protocols over the past few years.

5.1 Prioritized Random Access Protocols

This section reviews some of the priority schemes for data transmission, as well as integrated voice and data transmission, devised for random access communications channels. In general, for a priority scheme to be acceptable, it must possess the following attributes:

(1) The performance of a higher priority class should be immune to the presence of lower priority packets.

(2) Packets of the same priority class should be able to contend equally and fairly for channel bandwidth.

(3) The overhead required to implement a priority scheme must be kept at a minimal level.

Because of the widespread use of Ethernet [17], many priority schemes based on variations of the CSMA and CSMA/CD protocols were proposed [66-72, 82, 83].

Tobagi [82] proposed a message-based prioritized p-persistent CSMA protocol. In this protocol, a reservation period precedes channel contention periods. In the reservation period, each station with a packet to transmit makes a reservation. In the contention period, the highest priority class asking for channel access is granted exclusive transmission rights. Stations with packets belonging to this class use the p-persistent CSMA protocol for accessing the channel.

In [66], Maxemchuck devised a variation of CSMA/CD suitable for synchronous and asynchronous traffic. In this protocol, synchronous traffic, e.g., voice, is transmitted in a TDM-like manner and asynchronous traffic, e.g., data, is transmitted using CSMA/CD. Synchronous sources transmit using the CSMA protocol, i.e., sense the channel before transmission, but not while transmitting. Indeed, once a transmission from a synchronous source begins, it is never aborted. This is because packets from synchronous sources are preceded with a preamble that is "equal" in duration to the collision window. During the collision window, asynchronous sources that sense a collision will abort their transmissions before synchronous sources begin transmitting useful information. Collisions among synchronous sources are reduced by requiring them to schedule their next transmissions after fixed times.

Nutt and Bayer [67] studied the effect of different backoff algorithms (used to resolve collisions). The backoff algorithms considered were:

(a) Random: Backoff delay obtained via a uniform distribution.

(b) Binary exponential backoff: Same as Ethernet.

(c) Random voice/binary exponential data: Voice packets backoff delay is obtained via a uniform distribution such that they are not scheduled after their deadlines.

(d) Voice biased/binary exponential data: This algorithm favors voice packets over data packets.

The authors found that the binary exponential backoff algorithm is not well suited to combined voice and data. Their results also suggest that the random voice/binary exponential data algorithm is the best in terms of the voice packets loss. Their final point was that CSMA/CD networks are appropriate for carrying voice packets, provided that the network does not become overloaded.

Chlamtac and Eisenger [68] suggested a few variations to the Ethernet backoff algorithm in order to enhance the performance of integrated voice and data transmission. Such suggestions include increasing the first retransmission interval to give better randomization, and ensuring the placement of voice traffic into non-contending slots. They also suggested the removal of old voice packets, hence reducing the contention facing new packets, at the expense of some packet loss.

In [69], Goel and Elhakem suggested the use of a hybrid scheme in which data packets use the CSMA/CD protocol and voice packets are transmitted using a different protocol called FARA/CS (Frame Adaptable Reservation ALOHA with Carrier Sense). In FARA/CS, time is slotted and users with voice packets reserve a slot before the next frame. Slots are either reserved, idle or collided. A newly active or a collided user transmits in one of the idle or collided slots. Users with reservation, transmit in their reserved slots only. Other users are not to use reserved slots.

In [70], a variation of virtual time CSMA (VT-CSMA) called Reservation VT-CSMA (R-VT-CSMA), was proposed. In R-VT-CSMA, voice stations appear to have a dedicated TDM slot, and the delay of voice packets is bounded by the length of the frame (defined as the period between two successive voice packets from the same station). During a talkspurt, a voice station always schedules its next packet transmission at a fixed time within the frame (similar to [66]). Data packets are assigned a lower priority and operate according to the normal VT-CSMA protocol.

Other Prioritized random access protocols have been proposed for data transmission [83], as well as integrated voice and data transmission [71, 72]

In the following section, it is shown that implementing priorities on CASLANs can be achieved without the need for proposing new protocols. Indeed, it is shown that priority can be effectively implemented by simply varying node retry time values.

5.2. CASLANs with Two Priority Classes

In this section, CASLANs with two classes are considered, each of which has its own mean packet delay, arrival rate and retry time. A simulation model which is similar to that in section 2.1 is used for the study. The effect of the retry time on the average delay of each class is analyzed. Also the use of retry times in priority assignment is shown.

Let C_1 and C_2 denote classes 1 and 2, respectively. Let N_i, T_i, λ_i and τ_i respectively be the number of nodes, packet length in bits, arrival rate per bit time and the retry time in bits for C_i, where $i = 1$ or 2. First, the fixed packet length case, $T_1=T_2=T$ is studied. For simplicity, it is assumed that $\lambda_1=\lambda_2=\lambda$.

Figures 5.1 and 5.2 show the average delay versus τ_2 (with $N_1=N_2=10$, and T = 480 bits), for various values of λ at $\tau_1=$ 110 bits (a non-factor of T) and 120 bits (a factor of T) respectively. By analyzing these results, as well as results at different values of T, N, λ_1 and λ_2, the following observations can be made:

1) At light load, the higher the retry time, the higher the average delay. Subsequently, the class with the lower retry time has higher priority.

2) At high load, increasing the retry time for either class does not always mean higher average delay. Indeed, at retry time values that are factors of the packet length, a decrease in average delay in the form of a local minima is observed. This agrees with the single class results (see section 2.1).

3) The observed decrease in average delay of a class of users is associated with an increase in average delay of the other class, even though its retry time has not

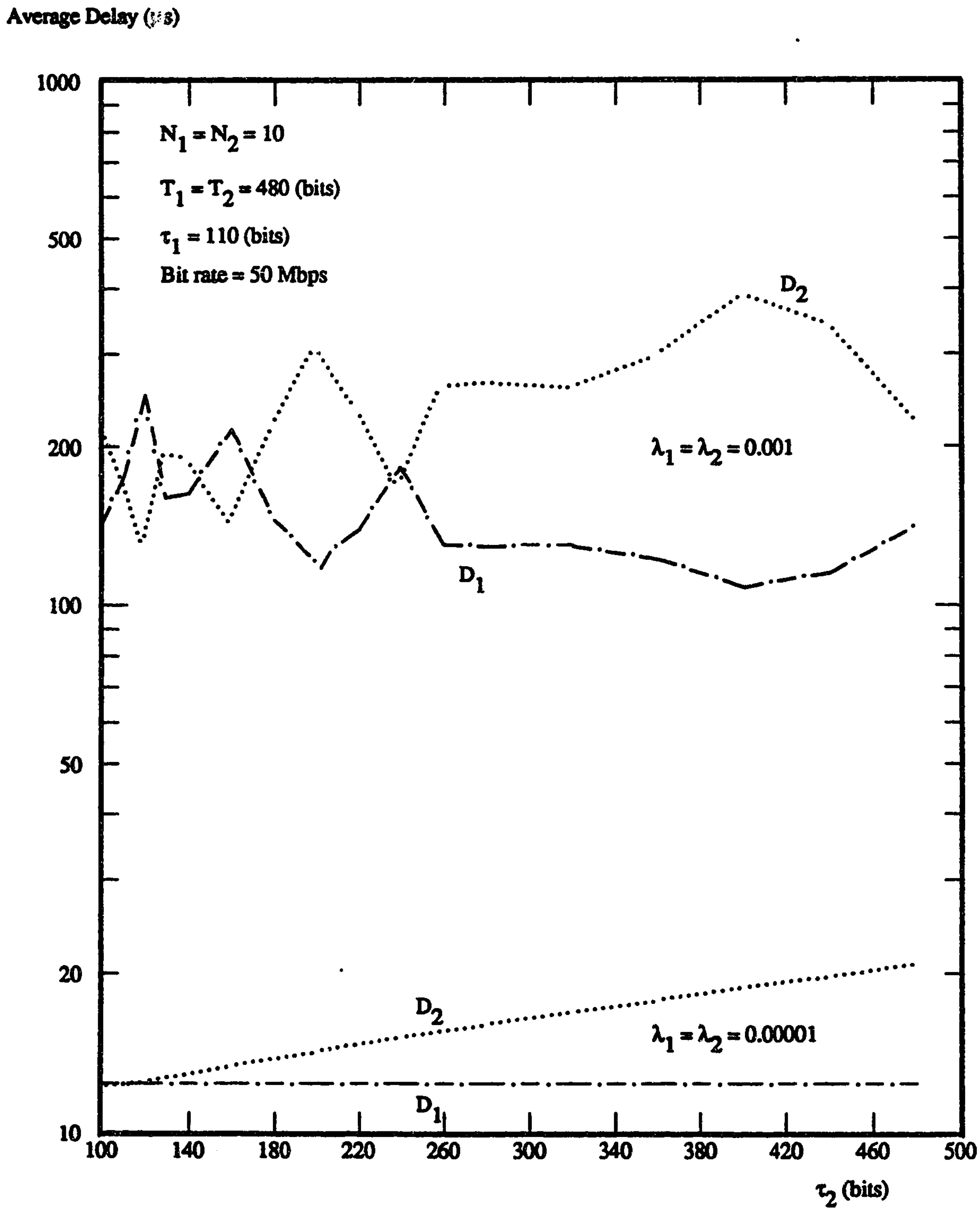


Figure 5.1: Response time at different values of τ_2 in a 20 node system.

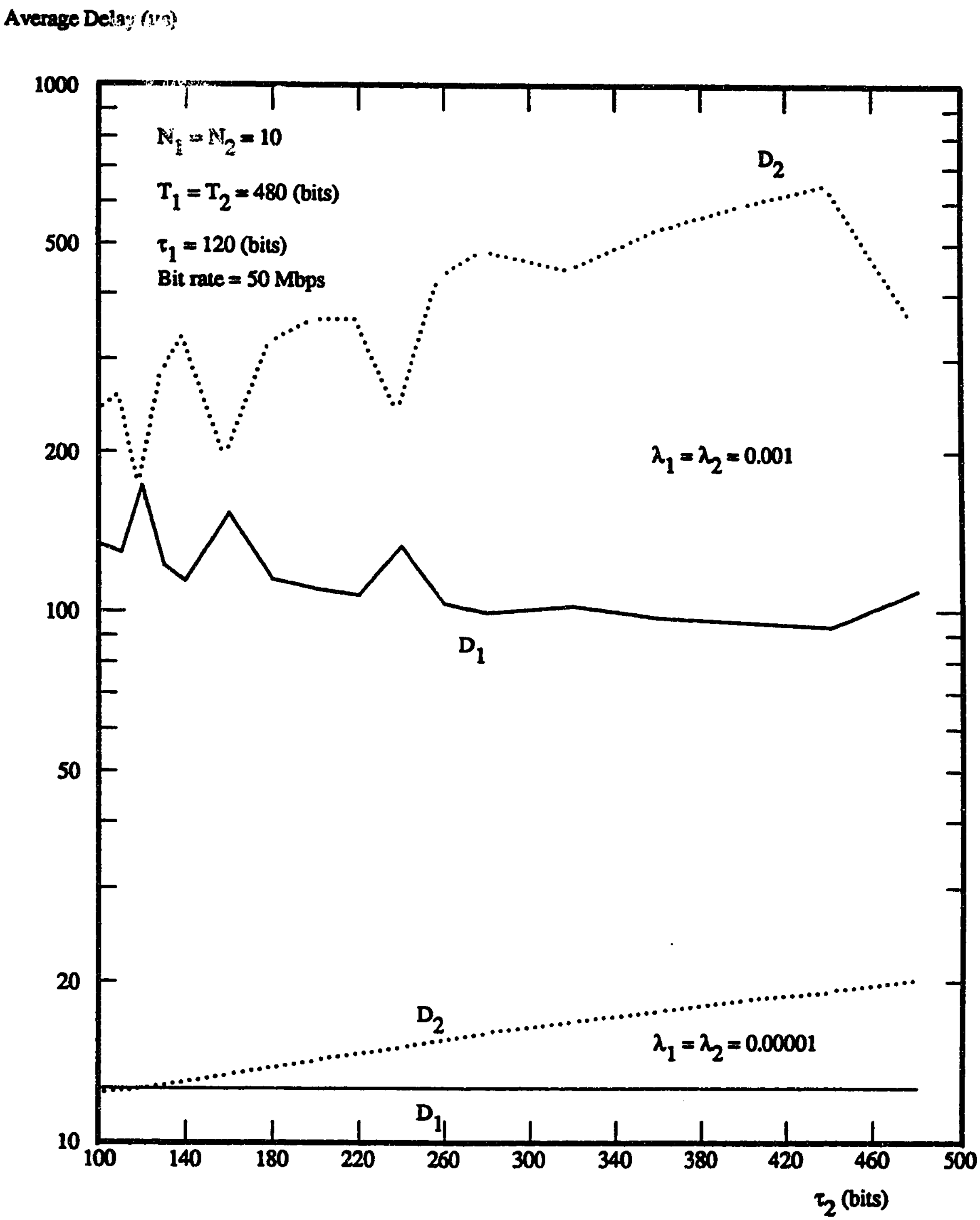


Figure 5.2: Response time at different values of τ_2 in a 20 node system.

changed.

Based on the above results, the following proposition is introduced.

Proposition 5.I:

Let τ_1 be a factor of T, and let $D_1(\tau)(D_2(\tau))$ denote the average delay of $C_1(C_2)$ at retry time τ. Then, there exists a period $[\tau_1-\delta,\infty)$ in which $D_1(\tau_1) \le D_2(\tau_2)$, where $\tau_2 \in [\tau_1-\delta,\infty)$.

□

The value of δ increases with the load and reduces to 0 at light load. At light load, $D_1(\tau_1) \le D_2(\tau_2)$ if and only if $\tau_1 \le \tau_2$. On the other hand, the range of τ_f (a factor of T) could, at heavy load, extend to the greatest factor of T less than τ_f. Note that the above proposition suggests that if τ_1 is a factor of T, then there exists no $\tau_2 > \tau_1$ such that C_2 would have a higher priority over C_1, i.e., lower delay. This proposition is equivalent to proposition 2.I for the single class case. Proposition 5.I, however, is concerned with comparing the retry times of two contending classes of users. On the other hand, proposition 2.I compares two modes of system operation, each under a different retry time value. The following results follow from proposition 5.I.

Corollary 5.1:

If τ_2 is a factor of T, then C_1 could only have higher priority than C_2 if $\tau_1 < \tau_2$.

Proof:

Since τ_2 is a factor of T, then its range includes all values higher than itself. Thus, for C_1

to have higher priority over C_2, τ_1 must be less than τ_2.

□

Corollary 5.2:

If τ_2 is a factor of T, then (in the absence of information about δ) the largest τ_1 value to be assigned to C_1 and still guarantee it higher priority over C_2 is equal to the greatest factor of T less than τ_2.

Proof:

Let τ be the assigned value for τ_1. If τ is a factor of T then its range would include every value greater than itself. Let τ_g be the greatest factor of T less than τ_2. If $\tau = \tau_g$, then τ_2 is in the range of τ and C_1 would have higher priority. Now, let τ, $\tau_g < \tau < \tau_2$ be a non-factor of T. Since the range of τ_2, $[\tau_2-\delta, \infty)$, may include τ, then assigning $\tau_1 = \tau$ does not guarantee a higher priority for C_1. Therefore, assigning τ_2 to the greatest factor of T that is less than τ_2 guarantees C_1 a higher probability over C_2.

□

In order to empirically verify the above corollaries, consider Figures 5.1 and 5.2. In Figure 5.1 with τ_1=110 bits, λ=0.001 and τ_2=160, $D_2(\tau_2) < D_1(\tau_1)$. In Figure 5.2, τ_1=120 bits and at the same values of λ and τ_2, it is found that $D_1(\tau_1) < D_2(\tau_2)$. That is, by increasing τ_1 from 110 to 120, C_1 is assigned a priority that is higher than that of C_2. This result also shows that $\tau_1 < \tau_2$, does not guarantee a higher priority for C_1. Only by making τ_1 a factor of T (τ_1= 120), is priority over C_2 guaranteed. Also note that with τ_1 = 120, C_1 always has lower average delay than C_2 for τ_2 in the range [100, 480], see Figure

5.2. In fact, the range of $\tau_1 = 120$, at offered load = 9.6, extends almost to the greatest lower factor retry time value (96) and is approximately given by [96.3, ∞).

It should be noted that the effect of changing the retry times is much more apparent here than in the single class case (section 2.1). For instance, in Figure 5.1, changing τ_2 from 240 to 180 at λ=0.001 yields an increase of 105% in average delay (D_2), as opposed to an increase of only 10% in the single class case. The reason is that as τ_2 changes such that C_2 has lower priority, C_1 packets will face less contention at the hub causing a decrease in $D_1(\tau_1)$ and C_2 packets will face more C_1 packets, and this causes more C_2 packet retries. This results in more C_1 packets being transmitted. This phenomenon, where one class monopolies the hub, is called the *hub hogging effect*. To illustrate the effect of hub hogging, the number of successfully transmitted packets of C_1 and C_2 are compared at different values of τ_1 and τ_2, see Table 5.1.

τ_1/τ_2 (bits)	No. of packets of C_1	No. of packets of C_2
110/ 140	26,350	23,650
110/ 160	20,469	29,531
110/ 280	32,130	17,870
120/ 140	35,791	14,209
120/ 160	27,620	22,380
120/ 280	40,559	9,441

Table 5.1: The hub hugging effect

In Table 5.1, λ=0.001, T=480 and a total of 50,000 packets were successfully transmitted over the simulation time. From the table it can be seen that hub hogging is strongest at $\tau_1(\tau_2)$ values which are factors of T. For instance, at τ_1=120 and τ_2=280, a total of 40,559 C_1 packets and only 9,441 C_2 packets are transmitted. With τ_1 = 110 and τ_2 = 280, these numbers changed dramatically to 32,130 packets from C_1 and 17,870 packets from C_2.

An implication of the apparent increase of hub hogging at retry time values that are factors of the packet length, is that a priority class using such a retry time is not severely affected by the presence of packets from another class that uses a non-factor retry time value. By exam[illegible] at offered load = 9.6, with $N_1 = N_2 =$ 10 and with τ_2 in the range [280, 440] [illegible]2- it can be seen that the average delay is very close to that at $N_2 = 0$ (single class CASLANs with N = 10) where D(120) = 81.5μs.

It was shown that assigning a factor retry time value to a priority class results in lower average packet delays. In some cases such as real-time applications, as discussed in chapter 4, reasonable average performance does not suffice. In real-time application, the probability of packets being received before their deadlines is the primary performance measure. This makes controlling the maximum delay as important as, and may be more important than, maintaining a low average delay. In Figure 5.3, maximum observed delays of C_1 (real-time traffic) versus τ_1 at various values of τ_2 (where C_2 is a non-real-time traffic) is shown, with $N_1 = N_2 = 10$, T = 480 and an offered load of 9.6.

From the results in Figure 5.3, it is not hard to notice the significantly lower maximum delay at retry time values that are factors of the packet length (120, 160, and 240). For instance, at $\tau_2 = 140$, the maximum delay of C_1 decreased from 2460 μs to 560 μs when τ_1 was increased from 100 to 120. Also note that a low maximum delay is maintained at retry time values that are factors of the packet length, even with τ_2 a factor of the packet length, see the delay at $\tau_2 = 160$ in Figure 5.3.

Finally, the case where $T_2 \neq T_1$ is studied. The distribution of T_2 is arbitrarily chosen to be exponential, which eliminates the effect of T_2 on the priority assignment. In Figure 5.4, the average delays of C_1 and C_2 are shown versus the retry time of C_1 (with τ_2=110 bits, N_1=N_2=10, and $T_1 = \bar{T}_2 = 480$). By analyzing of the results of Figure 5.4, the following observations are made.

Proposition 5.I and corollaries 5.1 and 5.2 still apparently hold in the case where $T_1 \neq T_2$. However, the effect of the factor retry times is not so profound. A decrease of only 21.5% is noted in D_1 when decreasing τ_1 from 140 to 120 at λ=0.001 and τ_2=110 bits, whereas in the fixed packet length case the decrease was 39%. This is due to the fact that $T_2 \neq T_1$, and that T_2 is variable causing the interdependence between the packet length and retry time to decrease. However, this interdependence did not completely vanish because T_1 is still fixed and τ_1 is a factor of T_1. Therefore, this still gives C_1 higher priority over C_2 within the range of τ_1.

The results in Figures 5.3 and 5.4 are of extreme importance, since they show the potential for using CASLANs for integrated services applications (e.g., voice and data). Since voice packets are of fixed length, they could be transmitted with a retry time that is a factor of the packet length. Variable length data packets could be transmitted with a retry time, a non-factor of the packet length that is greater than the retry time for voice

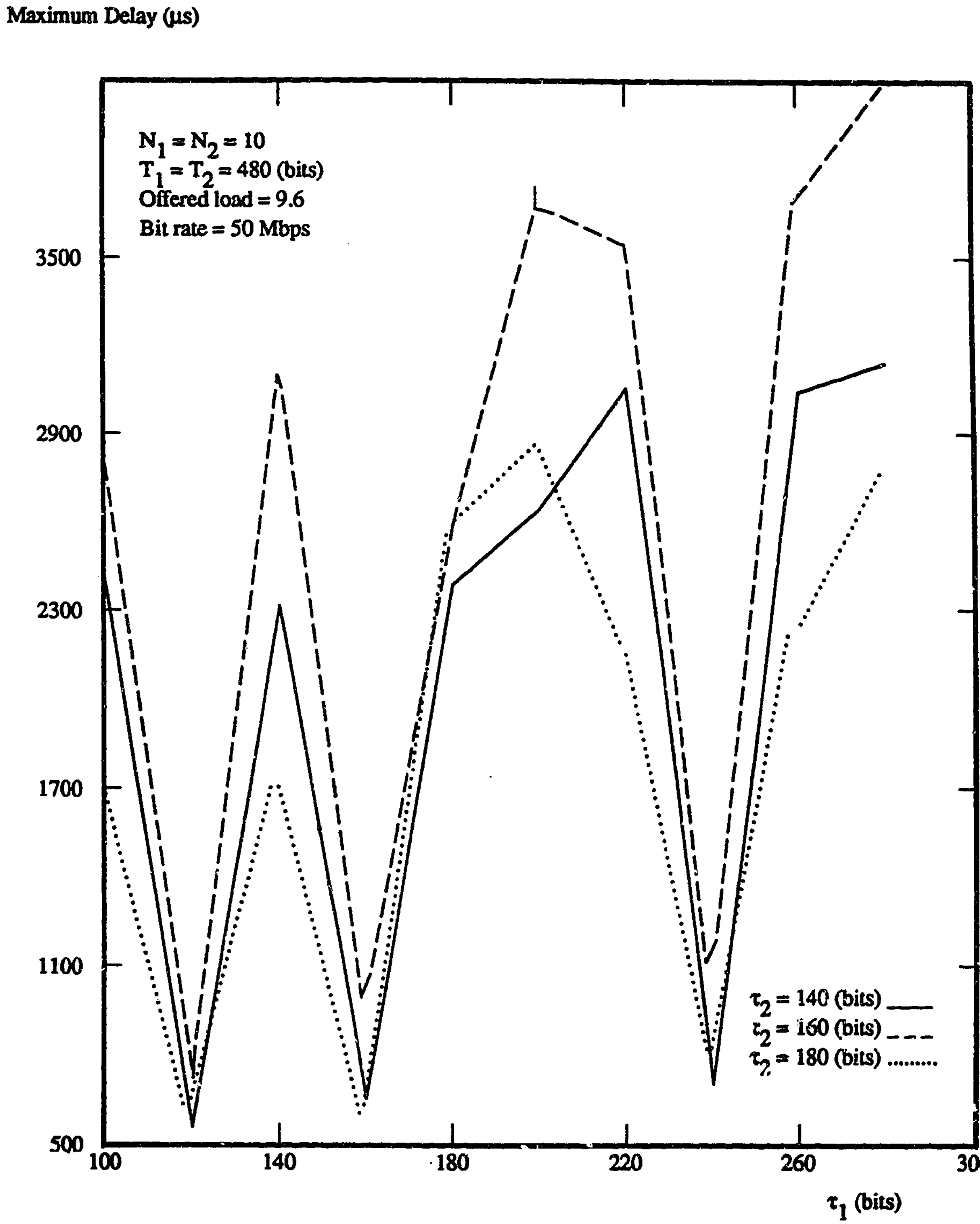


Figure 5.3: Maximum delay at different values of τ_1 in a 20 node system.

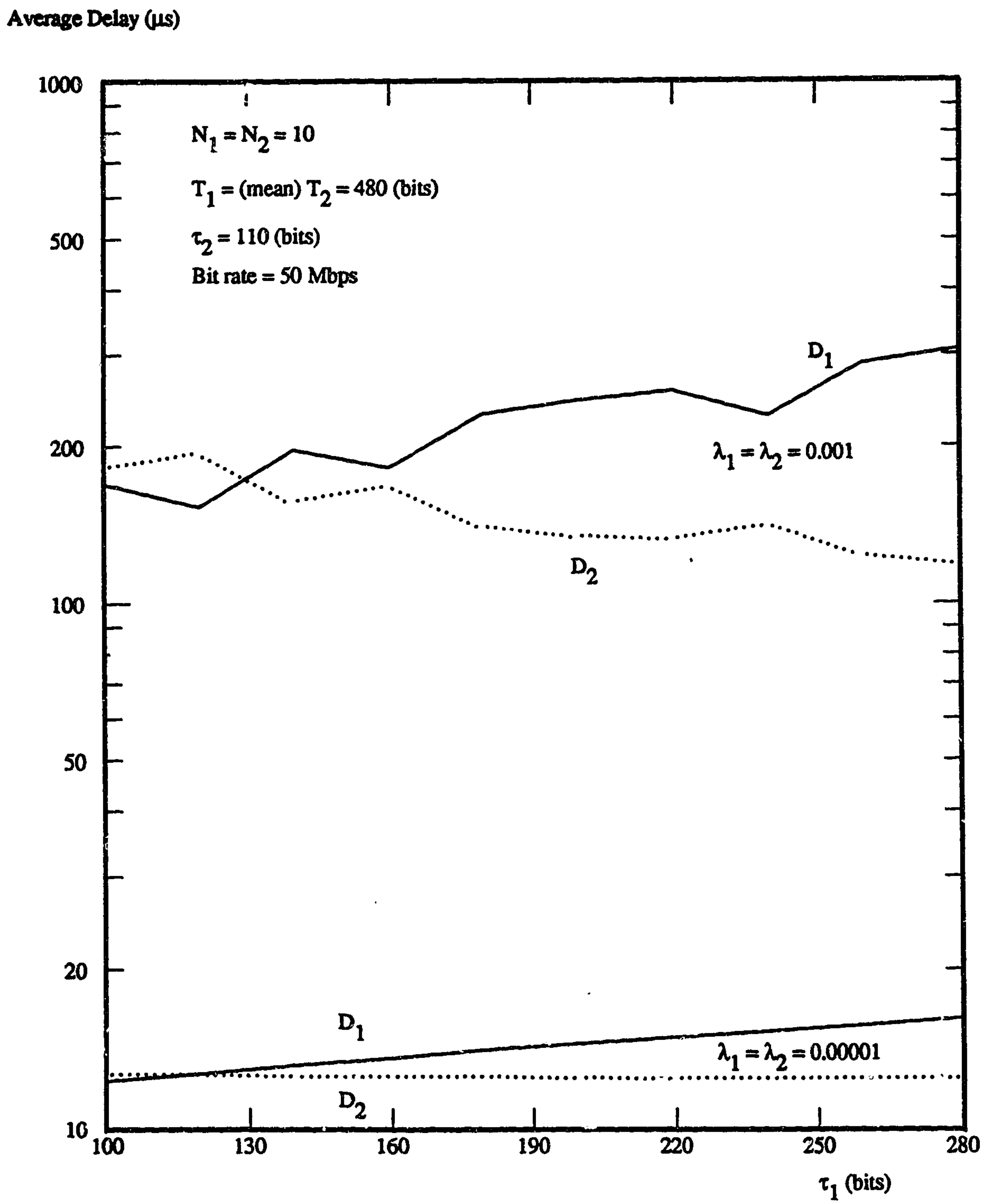


Figure 5.4: Response time at different values of τ_1 in a 20 node system.

packets. This would not only assign higher priority to voice packets, but also results in a very low percentage of lost voice packets. This is because of the much lower maximum delay associated with retry time values that are factors of the packet length. This issue is deferred to section 5.5.

5.3 Modeling of Prioritized CASLANs

In chapter 3, a performance model for symmetric CASLANs was presented. The model was based on following a tagged user (in an exact analytical approach), while the rest of the users were modeled to follow the steady state behavior of the tagged user. This model was shown to outperform all other CASLANs performance models mainly because it does not assume independence between packet retry instants at the hub.

In this section, a performance model of a two-class CASLAN is introduced. The model is a generalization of the single class model in chapter 3. The system is modeled as a polling system where the hub chooses at random one of the users of class 1 or class 2. The users of either class are divided into the following groups, namely,

(1) idle users,

(2) first-time ready users,

(3) carry-over ready users and

(4) a tagged user

These groups were defined earlier in section 3.2.

The following assumptions are made:

(1) All nodes are equipped with single packet occupancy buffers.

(2) Identical and independent packet generation process at all nodes. The packet generation time is exponentially distributed with rate $\lambda_1(\lambda_2)$ for class 1 (class 2).

(3) The packet length for each class is fixed and is given by $T_1(T_2)$ for class 1(class 2).

(4) For each class, the retry time is the same and is given by $\tau_1(\tau_2)$ for class 1(class 2).

(5) Min $(T_1, T_2) \geq$ Max (τ_1, τ_2).

(6) Except for the tagged users, the retry instants for packets of all nodes are independent of each other and of the packet length.

As in the model in section 3.2, a discrete time approach is used, where the retry time is divided into a number of slots. Since there are two retry time values, the choice of the grid size is not obvious. Recall that, in the single-class model, the grid size represented the retry time in number of slots. A choice of the same grid size for both classes would result in a different slot size for each class. Therefore, a fixed value ζ is chosen to be the slot size and two different grid sizes $g_1(g_2)$, are used for class 1(class 2), where

$$g_1 = \left\lceil \frac{\tau_1}{\zeta} \right\rceil \quad \text{and} \quad g_2 = \left\lceil \frac{\tau_2}{\zeta} \right\rceil$$

The packet size, $T_1(T_2)$, for class 1(class 2) is also given in slots and is equal to $K_1\tau_1+\Delta_1(K_2\tau_2+\Delta_2)$, where

$$K_1 = \left\lfloor \frac{T_1}{\tau_1} \right\rfloor, \qquad K_2 = \left\lfloor \frac{T_2}{\tau_2} \right\rfloor$$

and

$$\Delta_1 = T_1 \bmod \tau_1, \qquad \Delta_2 = T_2 \bmod \tau_2$$

An idle user may generate a packet in a slot with probability $\sigma_1(\sigma_2)$ for class 1(class 2), where

$$\sigma_1 = 1 - e^{-\lambda_1 \zeta} \quad \text{and} \quad \sigma_2 = 1 - e^{-\lambda_2 \zeta},$$

where $\lambda_1(\lambda_2)$ is the packet arrival rate per bit time for class 1(class 2).

A five-dimensional embedded Markovian chain is used to represent the polling system where the embedding points are at the start of hub acquisitions.

(1) The first dimension is the state in which the tagged users in both classes are in; namely $\bar{R}$ (not ready), A (acquiring the hub), and R (ready). Eight different combinations exist.

— $\bar{R}\bar{R}$ both tagged users are not ready.

— $\bar{R}A$ class 1 tagged user is not ready and class 2 tagged user is acquiring the hub.

— $\bar{R}R$ class 1 tagged user is not ready and class 2 tagged user is ready.

— $A\bar{R}$ class 1 tagged user is acquiring the hub and class 2 tagged user is not ready.

— AR class 1 tagged user is acquiring the hub and class 2 tagged user is ready.

— $R\bar{R}$ class 1 tagged user is ready and class 2 tagged user is not ready.

— RA class 1 tagged user is ready and class 2 tagged user is acquiring the hub.

— RR both tagged users are ready.

The AA case is non-existent, since only one user can acquire the hub at a time.

(2,3) The second and third dimension are the number of carry-over ready users from class 1 and class 2 respectively.

(4,5) The forth and fifth dimensions are the remainder of the hub acquisition time at the instant of arrival of the tagged user packet (Y) for class 1 and class 2 respectively. If any of the tagged users is not ready, then its corresponding Y value does not represent useful information and is, therefore, arbitrarily set to zero.

Depending on the state of both tagged users before and after the transition, 72 different transition probability cases exist, as opposed to only 9 for the single-class model in chapter 3. The state space for the Markovian chain consist of $(g_1+1)N_1(2N_2+1+g_2N_2)+(N_1+1)(g_2+1)N_2$ states, which is $O(N_1 \cdot N_2 \cdot g_1 \cdot g_2)$.

Because of the involvement and the computational expense of such a model, it is not pursued any further. Instead, a much simpler and computationally less expensive model is chosen. This model is described in the following section.

5.4. Light Load Prioritized CASLANs Performance Model

In this section, a two-class prioritized CASLAN protocol is analyzed. The retry time of a node is used for priority assignment. A 3-dimensional Markovian chain is used to model CASLANs which are represented by a polling system in which the central node chooses at random one of the idle or ready nodes. The model does not follow a tagged user and, in effect, does not keep track of interpacket arrival times of any user. Subsequently, such a model would not be able to capture the superior performance of retry time values that are factors of the packet length. Since such superiority is only apparent at high to heavy offered loads, the model is not expected to be very accurate at such loads. The model proposed in this section, thereupon, is intended for CASLANs that are under light to medium load conditions.

5.4.1. The Model

The model described here is an approximate model which is a generalization of an earlier model for the single class case [45]. The system is modeled as a polling system where the central hub chooses a node at random from the idle or ready nodes. The packet generation time at nodes is modified to be an exponential plus constant process, rather than exponential only, in order to reflect the effect of the propagation delay.

Let C_1 and C_2 be the two classes considered in the model. Let $N_1(N_2)$, $T_1(T_2)$, $\lambda_1(\lambda_2)$ and $\tau_1(\tau_2)$ be the number of nodes, packet length, arrival rate per time unit (bit time) and the retry time for $C_1(C_2)$. Let $\tau_1 \leq \tau_2$, thus giving higher priority to C_1 users. It is also assumed that $\text{Min}(T_1,T_2) \geq \text{Max}(\tau_1,\tau_2)$.

As in [45], the remainder of the retry times from ready nodes are modeled to obey a uniform distribution, where the arrival instant at the hub of the start of a packet transmission is chosen randomly from the period $(0, \tau_1(\tau_2)]$ for $C_1(C_2)$ nodes, following the end of a transmission. An implied assumption here is that a retry attempt does not depend on the original arrival time of the packet nor on previous retries. In actual CASLAN protocols, this is not the case.

The status of the hub is either serving a node (a packet transmission) or waiting for a packet. The latter state is called the waiting period. Because of the retry nature of the protocol at hand, the waiting period is no more than τ_1 if at least one node of C_1 is ready, no more than τ_2 if at least one node of C_2 is ready, and unbounded if there are no ready nodes.

A three-dimensional embedded Markovian chain is used to represent the polling system. The first two dimensions are the number of ready users in each class, i_1 and i_2, respectively, while the third is the last class to release the hub, i.e. last from $C_1(C_2)$.

The need to know the class that released the hub last follows from the fact that it is essential to know the retry time of the last packet to release the hub. This is because it takes at least τ ($\frac{\tau}{2}$ propagation time after the hub release from the hub to the node, and $\frac{\tau}{2}$ propagation time before hub acquisition from the node to the hub) for the node to release the hub to be able to acquire it again, where τ is $\tau_1(\tau_2)$ if the releasing node is from $C_1(C_2)$. Since τ_1 is different from τ_2, the third dimension representing the last class to release the hub is essential. If l_k denotes last from class k (where k = 1(2) for $C_1(C_2)$, then the state space could be represented by (i_1, i_2, l_k). One could then obtain the transition probabilities $P(j_1, j_2, l_m \mid i_1, i_2, l_k)$ from state (i_1, i_2, l_k) to state (j_1, j_2, l_m).

To keep notations compact, the following are defined for either class.

- $$P_{tx}(M,m,T,\lambda)=\begin{cases}\binom{M}{m}(1-e^{-\lambda T})^m\,(e^{-\lambda T})^{M-m} & 0\le m\le M\\ 0 & otherwise\end{cases}$$

This is the probability of m packet generations from M idle nodes during the transmission time (T) at arrival rate λ.

- $$P_{\bar{L}}(\tau,T,t,\lambda)=\begin{cases}e^{-\lambda(t+T-\tau)} & 0<t\le\tau\\ e^{-\lambda T} & t>\tau\end{cases}$$

This is the probability that the last node to acquire the hub will not reattempt transmission again during the next acquisition period following the hub release given that it occurred at time t.

- $$P_L(\tau,T,t,\lambda)=1-P_{\bar{L}}(\tau,T,t,\lambda)$$

- $$f(i,\tau,t)=\frac{i\,(\tau-t)^{i-1}}{\tau^i}$$

This is the probability that the first transmission from i ready users arrives at the hub at exactly time t. Since t, the overhead period, is undefined once the hub is acquired, such a transmission would then be the first to arrive at the hub from both idle or ready nodes. Subsequently, the packet would acquire the hub. In other words, this is the probability that one of i ready nodes would acquire the hub at exactly time t.

- $$P_s(M,t,\lambda)=\lambda Me^{-\lambda Mt}$$

This is the probability that one out of M idle nodes acquires the hub at time t.

- $P_{\bar{s}}(M,t,\lambda) = e^{-\lambda M t}$

This is the probability that none of the M idle nodes acquires the hub before time t.

Similarly, the following are defined,

- $$P_s(M,\tau,t,\lambda) = \begin{cases} P_s(M-1,t,\lambda) & 0<t\le\tau \\ P_s(M-1,t,\lambda)\cdot e^{-\lambda(t-\tau)} + P_{\bar{s}}(M-1,t,\lambda)\cdot\lambda e^{-\lambda(t-\tau)} & t>\tau \end{cases}$$

and

- $$P_{\bar{s}}(M,\tau,t,\lambda) = \begin{cases} P_{\bar{s}}(M-1,t,\lambda) & 0<t\le\tau \\ P_{\bar{s}}(M-1,t,\lambda)\cdot e^{-\lambda(t-\tau)} & t>\tau \end{cases}$$

Notice that all of the above probabilities are joint on t, the overhead or waiting time. Since t is dependent on the number of ready users in the system, the joint transition probabilities $P^{(t)}(j_1,j_2,l_k \mid i_1,i_2,l_m)$ must be obtained first then an integration over t should be performed. Four different cases of transition probabilities exist depending on the class of the last user to acquire the hub.

<u>case $l_1 \rightarrow l_1$:</u>

$$P^{(t)}(j_1,j_2,l_1 \mid i_1,i_2,l_1) = P_{\bar{s}}(N_2-i_2,t,\lambda_2)\cdot(1-F(i_2,\tau_2,t))\cdot P_{\alpha}(N_2-i_2,j_2-i_2,T_1,\lambda_2)$$

$$\cdot\left\{P_{\bar{s}}(N_1-i_1,\tau_1,t,\lambda_1)\cdot f(i_1,\tau_1,t)\cdot\begin{bmatrix}P_{\bar{L}}(\tau_1,T_1,t,\lambda_1)\cdot P_{\alpha}(N_1-i_1-1,j_1-i_1+1,T_1,\lambda_1)+\\P_L(\tau_1,T_1,t,\lambda_1)\cdot P_{\alpha}(N_1-i_1-1,j_1-i_1,T_1,\lambda_1)\end{bmatrix}\right.$$

$$\left.+P_s(N_1-i_1,\tau_1,t,\lambda_1)\cdot(1-F(i_1,\tau_1,t))\cdot\begin{bmatrix}P_{\bar{L}}(\tau_1,T_1,t,\lambda_1)\cdot P_{\alpha}(N_1-i_1-2,j_1-i_1,T_1,\lambda_1)+\\P_L(\tau_1,T_1,t,\lambda_1)\cdot P_{\alpha}(N_1-i_1-2,j_1-i_1-1,T_1,\lambda_1)\end{bmatrix}\right\} \quad (5.1)$$

The common multiplier outside the curly brackets in equation (5.1) above is the probability that none of C_2 users acquires the hub and that j_2-i_2 idle users from C_2 become ready during T_1. The term inside the curly bracket is the probability that a user from C_1 acquires the hub and that the number of ready users from C_1 by the end of the transmission is equal to j_1.

<u>case $l_2 \rightarrow l_1$:</u>

$$P^{(t)}(j_1,j_2,l_1 \mid i_1,i_2,l_2)=P_{\bar{s}}(N_2-i_2,t,\lambda_2)\cdot(1-F(i_2,\tau_2,t))$$

$$\times\begin{bmatrix}P_{\bar{L}}(\tau_2,T_1,t,\lambda_2)\cdot P_{\alpha}(N_2-i_2-1,j_2-i_2,T_1,\lambda_2)+\\P_L(\tau_2,T_1,t,\lambda_2)\cdot P_{\alpha}(N_2-i_2-1,j_2-i_2-1,T_1,\lambda_2)\end{bmatrix}$$

$$\times\begin{bmatrix}P_{\bar{s}}(N_1-i_1,\tau_1,t,\lambda_1)\cdot f(i_1,\tau_1,t)\cdot P_{\alpha}(N_1-i_1,j_1-i_1+1,T_1,\lambda_1)+\\P_s(N_1-i_1,\tau_1,t,\lambda_1)\cdot(1-F(i_1,\tau_1,t))\cdot P_{\alpha}(N_1-i_1,j_1-i_1,T_1,\lambda_1)\end{bmatrix} \quad (5.2)$$

The first term of equation (5.2) above is the probability that none of C_2 users acquires the hub. The second term is the probability that j_2-i_2 idle users from C_2 become ready during T_1. The third term is the probability that a user from C_1 acquires the hub and that the number of ready users from C_1 by the end of the transmission is equal to j_1.

<u>case $l_1 \to l_2$:</u>

This case is similar to the $l_2 \to l_1$ case with all the information for C_1 and C_2 interchanged.

$$P^{(t)}(j_1,j_2,l_2 \mid i_1,i_2,l_1) = P_{\bar{s}}(N_1-i_1,t,\lambda_1)\cdot(1-F(i_1,\tau_1,t))$$

$$\times \begin{bmatrix} P_{\bar{L}}(\tau_1,T_2,t,\lambda_1)\cdot P_{tx}(N_1-i_1-1,j_1-i_1,T_2,\lambda_1)+ \\ P_L(\tau_1,T_2,t,\lambda_1)\cdot P_{tx}(N_1-i_1-1,j_1-i_1-1,T_2,\lambda_1) \end{bmatrix}$$

$$\times \begin{bmatrix} P_{\bar{s}}(N_2-i_2,\tau_2,t,\lambda_2)\cdot f(i_2,\tau_2,t)\cdot P_{tx}(N_2-i_2,j_2-i_2+1,T_2,\lambda_2)+ \\ P_s(N_2-i_2,\tau_2,t,\lambda_2)\cdot(1-F(i_2,\tau_2,t))\cdot P_{tx}(N_2-i_2,j_2-i_2,T_2,\lambda_2) \end{bmatrix} \tag{5.3}$$

<u>case $l_2 \to l_2$:</u>

This case is similar to the $l_1 \to l_1$ case with all the information for C_1 and C_2 interchanged.

$$P^{(t)}(j_1,j_2,l_2 \mid i_1,i_2,l_2) = P_{\bar{s}}(N_1-i_1,t,\lambda_1)\cdot(1-F(i_1,\tau_1,t))\cdot P_{tx}(N_1-i_1,j_1-i_1,T_2,\lambda_1)$$

$$\cdot \left\{ P_{\bar{s}}(N_2-i_2,\tau_2,t,\lambda_2)\cdot f(i_2,\tau_2,t)\cdot \begin{bmatrix} P_{\bar{L}}(\tau_2,T_2,t,\lambda_2)\cdot P_{tx}(N_2-i_2-1,j_2-i_2+1,T_2,\lambda_2) + \\ P_L(\tau_2,T_2,t,\lambda_2)\cdot P_{tx}(N_2-i_2-1,j_2-i_2,T_2,\lambda_2) \end{bmatrix} \right.$$

$$\left. + P_s(N_2-i_2,\tau_2,t,\lambda_2)\cdot(1-F(i_2,\tau_2,t))\cdot \begin{bmatrix} P_{\bar{L}}(\tau_2,T_2,t,\lambda_2)\cdot P_{tx}(N_2-i_2-2,j_2-i_2,T_2,\lambda_2)+ \\ P_L(\tau_2,T_2,t,\lambda_2)\cdot P_{tx}(N_2-i_2-2,j_2-i_2-1,T_2,\lambda_2) \end{bmatrix} \right\} \tag{5.4}$$

To obtain the steady state probabilities, a transformation from the three dimensional state space to a one dimensional state space may be employed. In such a transformation, a state (i_1, i_2, l_k) would be mapped to state m_1 if l_k=1, or to m_2 if l_k=2, where

$$m_1 = i_1 \cdot N_1 + i_2$$

and

$$m_2 = N_1 \cdot (N_2+1) + i_1 \cdot N_1 + i_2$$

Note that the state space would consist of $N_1 \cdot (N_2+1) + (N_1+1) \cdot N_2$ states, i.e. $O(N_1 \cdot N_2)$.

The steady state probabilities can be obtained by solving the set of equations in (5.5) below

$$\vec{\pi} = \vec{\pi} \cdot P$$

and

$$\vec{\pi} \cdot \vec{1} = 1 \tag{5.5}$$

where P is the one-dimensional transition probability matrix and $\vec{\pi}$ is the row vector containing the steady state probabilities, and $\vec{1}$ represents a column vector of 1 elements.

5.4.2. Throughput and Average Delay

To obtain the throughput and average delay for each class, the regenerative nature of Markovian chains is used. The entrance to any state, say state $(0,0,l_1)$, is a regenerative process. Let R_{i_1, i_2, l_k} denote the expected time to reach state $(0,0,l_1)$ for the first time from state (i_1, i_2, l_k), and let $TX1_{i_1, i_2, l_k}$ and $TX2_{i_1, i_2, l_k}$ respectively denote the number of successful transmissions of class 1 and class 2 during R_{i_1, i_2, l_k}. If X_1 and X_2 respectively represent the throughput per time unit for class 1 and class 2, then X_1 and X_2 are given by equations (5.6) and (5.7) below.

$$X_1 = \frac{TX1_{0,0,l_1}}{R_{0,0,l_1}} \tag{5.6}$$

$$X_2 = \frac{TX2_{0,0,l_1}}{R_{0,0,l_1}} \tag{5.7}$$

The average delays for classes 1 and 2, D_1 and D_2, are respectively given by equations (5.8) and (5.9).

$$D_1 = \frac{N_1}{X_1} - \frac{1}{\lambda_1} \tag{5.8}$$

$$D_2 = \frac{N_2}{X_2} - \frac{1}{\lambda_2} \tag{5.9}$$

Now, $TX1_{0,0,l_1}$, $TX2_{0,0,l_1}$ and $R_{0,0,l_1}$ can be obtained by solving the set of simultaneous equations in (5.10), (5.11) and (5.12) below.

$$TX1_{0,0,l_1} = \sum_{i_1=0}^{N_1-1} \sum_{i_2=0}^{N_2} P(i_1,i_2,l_1 | 0,0,l_1) \cdot (1+TX1_{i_1,i_2,l_1})$$

$$+ \sum_{i_1=0}^{N_1} \sum_{i_2=0}^{N_2-1} P(i_1,i_2,l_2 | 0,0,l_1) \cdot (0+TX1_{i_1,i_2,l_2})$$

$$- TX1_{0,0,l_1} \cdot P(0,0,l_1 | 0,0,l_1) \tag{5.10a}$$

$$TX1_{i_1,i_2,l_1} = \sum_{j_1=0}^{N_1-1} \sum_{j_2=0}^{N_2} P(j_1,j_2,l_1 | i_1,i_2,l_1) \cdot (1+TX1_{j_1,j_2,l_1})$$

$$+\sum_{j_1=0}^{N_1}\sum_{j_2=0}^{N_2-1} P(j_1,j_2,l_2 \mid i_1,i_2,l_1)\cdot(0+TX1_{j_1,j_2,l_2})$$

$$-TX1_{0,0,l_1}\cdot P(0,0,l_1 \mid i_1,i_2,l_1) \qquad (5.10b)$$

$$TX1_{i_1,i_2,l_2}=\sum_{j_1=0}^{N_1-1}\sum_{j_2=0}^{N_2} P(j_1,j_2,l_1 \mid i_1,i_2,l_2)\cdot(1+TX1_{j_1,j_2,l_1})$$

$$+\sum_{j_1=0}^{N_1}\sum_{j_2=0}^{N_2-1} P(j_1,j_2,l_2 \mid i_1,i_2,l_2)\cdot(0+TX1_{j_1,j_2,l_2})$$

$$-TX1_{0,0,l_1}\cdot P(0,0,l_1 \mid i_1,i_2,l_2) \qquad (5.10c)$$

$$TX2_{0,0,l_1}=\sum_{i_1=0}^{N_1-1}\sum_{i_2=0}^{N_2} P(i_1,i_2,l_1 \mid 0,0,l_1)\cdot(0+TX2_{i_1,i_2,l_1})$$

$$+\sum_{j_1=0}^{N_1}\sum_{j_2=0}^{N_2-1} P(j_1,j_2,l_2 \mid 0,0,l_1)\cdot(1+TX2_{i_1,i_2,l_2})$$

$$-TX2_{0,0,l_1}\cdot P(0,0,l_1 \mid 0,0,l_1) \qquad (5.11a)$$

$$TX2_{i_1,i_2,l_1}=\sum_{j_1=0}^{N_1-1}\sum_{j_2=0}^{N_2} P(j_1,j_2,l_1 \mid i_1,i_2,l_1)\cdot(0+TX2_{j_1,j_2,l_1})$$

$$+\sum_{j_1=0}^{N_1}\sum_{j_2=0}^{N_2-1} P(j_1,j_2,l_2 \mid i_1,i_2,l_1)\cdot(1+TX2_{j_1,j_2,l_2})$$

$$-TX2_{0,0,l_1}\cdot P(0,0,l_1 \mid i_1,i_2,l_1) \qquad (5.11b)$$

$$TX2_{i_1,i_2,l_2}=\sum_{j_1=0}^{N_1-1}\sum_{j_2=0}^{N_2} P(j_1,j_2,l_1 \mid i_1,i_2,l_2)\cdot(0+TX2_{j_1,j_2,l_1})$$

$$+\sum_{j_1=0}^{N_1}\sum_{j_2=0}^{N_2-1} P(j_1,j_2,l_2 \mid i_1,i_2,l_2)\cdot(1+TX2_{j_1,j_2,l_2})$$

$$-TX2_{0,0,l_1}\cdot P(0,0,l_1 \mid i_1,i_2,l_2) \qquad (5.11c)$$

$$R_{0,0,l_1}=\sum_{i_1=0}^{N_1-1}\sum_{i_2=0}^{N_2} P(i_1,i_2,l_1 \mid 0,0,l_1)\cdot(T_1+R_{i_1,i_2,l_1})$$

$$+\sum_{i_1=0}^{N_1}\sum_{i_2=0}^{N_2-1} P(i_1,i_2,l_2 \mid 0,0,l_1)\cdot(T_2+R_{i_1,i_2,l_2})$$

$$-R_{0,0,l_1}\cdot P(0,0,l_1 \mid 0,0,l_1)+\overline{W}_{0,0,l_1} \qquad (5.12a)$$

$$R_{i_1,i_2,l_1}=\sum_{j_1=0}^{N_1-1}\sum_{j_2=0}^{N_2} P(j_1,j_2,l_1 \mid i_1,i_2,l_1)\cdot(T_1+R_{j_1,j_2,l_1})$$

$$+\sum_{j_1=0}^{N_1}\sum_{j_2=0}^{N_2-1} P(j_1,j_2,l_2 \mid i_1,i_2,l_1)\cdot(T_2+R_{j_1,j_2,l_2})$$

$$-R_{0,0,l_1}\cdot P(0,0,l_1 \mid i_1,i_2,l_1)+\overline{W}_{i_1,i_2,l_1} \qquad (5.12b)$$

$$R_{i_1,i_2,l_2}=\sum_{j_1=0}^{N_1-1}\sum_{j_2=0}^{N_2} P(j_1,j_2,l_1 \mid i_1,i_2,l_2)\cdot(T_1+R_{j_1,j_2,l_1})$$

$$+\sum_{j_1=0}^{N_1}\sum_{j_2=0}^{N_2-1} P(j_1,j_2,l_2 \mid i_1,i_2,l_2)\cdot(T_2+R_{j_1,j_2,l_2})$$

$$-R_{0,0,l_1}\cdot P(0,0,l_1 \mid i_1,i_2,l_2)+\overline{W}_{i_1,i_2,l_2} \qquad (5.12c)$$

where, $\overline{W}_{i_1,i_2,l_1}(\overline{W}_{i_1,i_2,l_2})$ is the average waiting period if the last node to release the hub is from $C_1(C_2)$, with i_1 ready users from C_1 and i_2 ready users from C_2. Expressions for the waiting period are given below

$$\overline{W}_{0,0,l_1}=\int_0^{\tau_1} t\cdot\Big[P_s(N_1-1,t,\lambda_1)\times P_{\bar{s}}(N_2,t,\lambda_2)+P_{\bar{s}}(N_1-1,t,\lambda_1)\times P_s(N_2,t,\lambda_2)\Big]\cdot dt$$

$$+\int_{\tau_1}^{\infty} t\cdot\Big\{P_{\bar{s}}(N_2,t,\lambda_2)\times\Big[P_s(N_1-1,t,\lambda_1)\times e^{-\lambda_1(t-\tau_1)}+P_{\bar{s}}(N_1-1,t,\lambda_1)\times\lambda_1 e^{-\lambda_1(t-\tau_1)}\Big]$$

$$+P_{\bar{s}}(N_1-1,t,\lambda_1)\times P_s(N_2,t,\lambda 2)\Big\}\cdot dt \qquad (5.13a)$$

$$\overline{W}_{0,i_2,l_1}=\int_0^{\tau_1} t\cdot\Big[P_s(N_1,t,\lambda_1)\times P_{\bar{s}}(N_2-i_2,t,\lambda_2)\times(1-F_2(i_2,t))$$

$$+P_{\bar{s}}(N_1-1,t,\lambda_1)\times P_s(N_2-i_2,t,\lambda_2)\times(1-F_2(i_2,t))+P_{\bar{s}}(N_1-1,t,\lambda_1)\times P_{\bar{s}}(N_2-i_2,t,\lambda_2)\times f_2(i_2,t)\Big]\cdot dt$$

$$+\int_{\tau_1}^{\tau_2} t\cdot\Big\{P_{\bar{s}}(N_2-i_2,t,\lambda_2)\times(1-F_2(i_2,t))\times\Big[P_s(N_1-1,t,\lambda_1)\times e^{-\lambda_1(t-\tau_1)}+P_{\bar{s}}(N_1-1,t,\lambda_1)\times\lambda_1 e^{-\lambda_1(t-\tau_1)}\Big]$$

$$+P_{\bar{s}}(N_1-1,t,\lambda_1)\times e^{-\lambda_1(t-\tau_1)}\times P_s(N_2-i_2,t,\lambda_2)\times(1-F_2(i_2-t))$$

$$+P_{\bar{s}}(N_1-1,t,\lambda_1)\times e^{-\lambda_1(t-\tau_1)}\times P_{\bar{s}}(N_2-i_2,t,\lambda_2)\times f_2(i_2,t)\Big\}\cdot dt \qquad (5.13b)$$

$$\overline{W}_{i_1,i_2,l_1}=\int_0^{\tau_1} t\cdot\Big[p_s(N_1-i_1-1,t,\lambda_1)\times(1-F_1(i_1,t))\times P_{\bar{s}}(N_2-i_2,t,\lambda_2)\times(1-F_2(i_2,t))$$

$$+P_{\bar{s}}(N_1-i_1-1,t,\lambda_1)\times f_1(i_1,t)\times P_{\bar{s}}(N_2-i_2,t,\lambda_2)\times(1-F_2(i_2,t))$$

$$+P_{\bar{s}}(N_1-i_1-1,t,\lambda_1)\times(1-F_1(i_1,t))\times P_s(N_2-i_2,t,\lambda_2)\times(1-F_2(i_2,t))$$

$$+P_{\bar{s}}(N_1-i_1,t,\lambda_1)\times(1-F_1(i_1,t))\times P_{\bar{s}}(N_2-i_2,t,\lambda_2)\times f_2(i_2,t)\Big]\cdot dt \quad (5.13c)$$

$$\overline{W}_{0,0,i_2}=\int_0^{\tau_2} t\cdot\Big[P_s(N_2-1,t,\lambda_2)\times P_{\bar{s}}(N_1,t,\lambda_1)+P_{\bar{s}}(N_2-1,t,\lambda_2)\times P_s(N_1,t,\lambda_1)\Big]\cdot dt$$

$$+\int_{\tau_2}^{\infty} t\cdot\Big\{P_{\bar{s}}(N_1,t,\lambda_1)\times\Big[P_s(N_2-1,t,\lambda_2)\times e^{-\lambda_2(t-\tau_2)}+P_{\bar{s}}(N_2-1,t,\lambda_2)\times\lambda_2 e^{-\lambda_2(t-\tau_2)}\Big]$$

$$+P_{\bar{s}}(N_2-1,t,\lambda_2)\times P_s(N_1,t,\lambda 1)\Big\}\cdot dt \quad (5.14a)$$

$$\overline{W}_{0,i_2,i_2}=\int_0^{\tau_2} t\cdot\Big[p_s(N_2-i_2-1,t,\lambda 2)\times(1-F_2(i_2,t))\times P_{\bar{s}}(N_1,t,\lambda_1)$$

$$+P_{\bar{s}}(N_2-i_2-1,t,\lambda_2)\times f_2(i_2,t)\times P_{\bar{s}}(N_1,t,\lambda_1)$$

$$+P_{\bar{s}}(N_2-i_2-1,t,\lambda_2)\times(1-F_2(i_2,t))\times P_s(N_1,t,\lambda_1)\Big]\cdot dt \quad (5.14b)$$

$$\overline{W}_{i_1,i_2,i_2}=\int_0^{\tau_1} t\cdot\Big[p_s(N_2-i_2-1,t,\lambda_2)\times(1-F_2(i_2,t))\times P_{\bar{s}}(N_1-i_1,t,\lambda_1)\times(1-F_1(i_1,t))$$

$$+P_{\bar{s}}(N_2-i_2-1,t,\lambda_2)\times f_2(i_2,t)\times P_{\bar{s}}(N_1-i_1,t,\lambda_1)\times(1-F_1(i_1,t))$$

$$+P_{\bar{s}}(N_2-i_2-1,t,\lambda_2)\times(1-F_2(i_2,t))\times P_s(N_1-i_1,t,\lambda_1)\times(1-F_1(i_1,t))$$

$$+P_{\bar{s}}(N_2-i_2,t,\lambda_2)\times(1-F_2(i_2,t))\times P_{\bar{s}}(N_1-i_1,t,\lambda_1)\times f_1(i_1,t)\Big]\cdot dt \quad (5.14c)$$

5.4.3. Numerical Results

In this section, some results obtained from the model presented in the previous two subsections are shown. These results are compared to results from the simulation model used in section 5.2.

Figures 5.5 and 5.6 show the average delay of C_1 and C_2 users as a function of τ_2 for a 4-node and a 20-node systems respectively. Note that increasing the retry time results in an increase in average delay associated with a decrease in throughput. Moreover, by fixing τ_1 and increasing τ_2, the delay for class 1 packets decreases even though τ_1 did not decrease. This is because class 1 packets will face less contention at the central hub as τ_2 increases. The above phenomenon is even more apparent at relatively high load. For instance, in Figure 5.6, a decrease of 42.7% in average delay for class 1 is observed if τ_2 is to be increased form 120 to 480 (bits) at an arrival rate of 0.0004, while a decrease of only 3% is observed at an arrival rate of 0.00004.

Figure 5.7 shows the average delay of C_1 and C_2 users versus τ_2 in a 4 node and a 12 node systems under heavy traffic with $T_2 = 2 \cdot T_1$. By increasing τ_2, priority is given to C_1 users, i.e., priority is for short packets. Note the sharp decrease in average delay for C_1 users with the increase of τ_2 (54.4% over the range 120-480 for the 12 node system).

Simulation results show that the model yields very accurate results at light to medium offered load conditions. At heavy load, however, there exists some points where the analytical results are not very accurate, see Figures 5.5 and 5.6. This was expected, however, since the model does not keep track of packets retry instants at the hub.

5.5. Voice and Data Integration

Integration of voice and data on the same network has received considerable attention in the past few years. With the possibility of voice digitization and packetization,

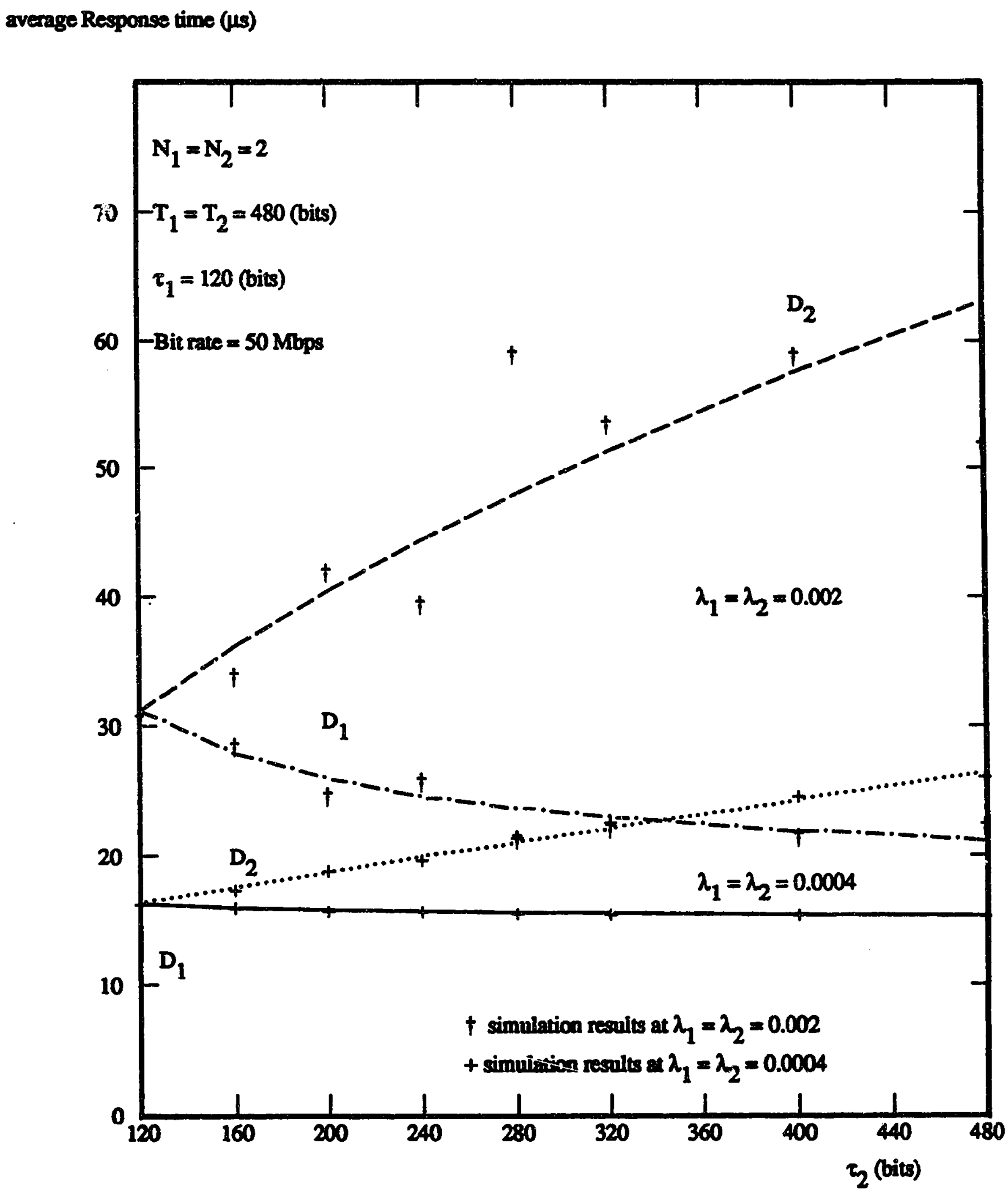


Figure 5.5: Response time at different values of τ_2 in a 4 node system.

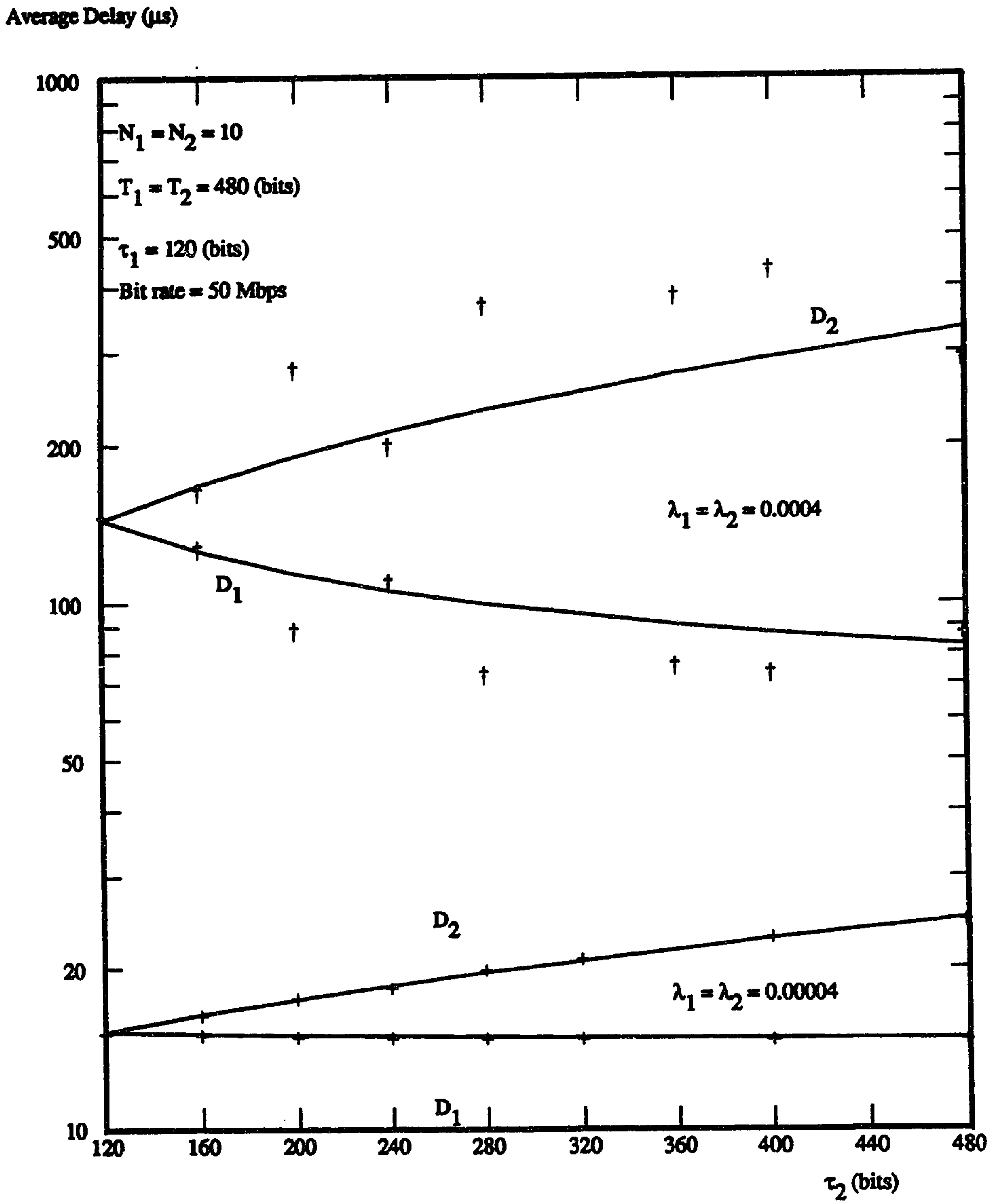


Figure 5.6: Response time at different values of τ_2 in a 20 node system.

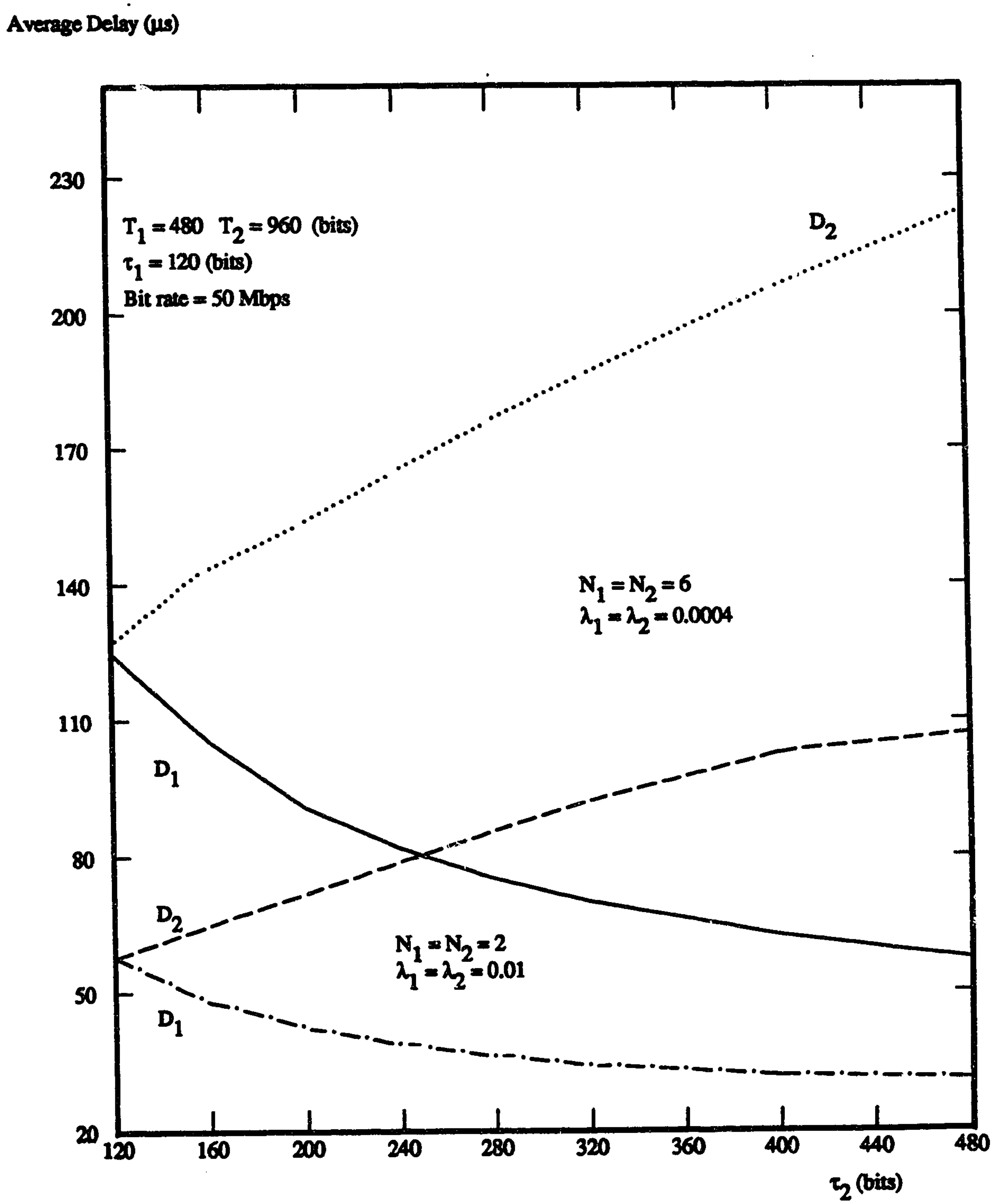


Figure 5.7: Response time at different values of τ_2 with $T_2 = 2T_1$.

integration of voice and data seems more appealing from the economic and practical points of view [51, 84]. In this section, integrated voice and data CASLANs are studied. A simulation model is introduced from which the performance of integrated voice and data CASLANs is presented. It is shown that by properly setting CASLANs network parameters, voice packet loss could be eliminated, even in large systems with heavy data offered loads. First, a review of voice traffic and modeling concepts is presented.

5.5.1 Background

Characteristics of voice traffic:

A typical behavior of a voice source, which generates packets from a voice signal is illustrated in Figure 5.8. A voice source is *active* when the talker is actually speaking.

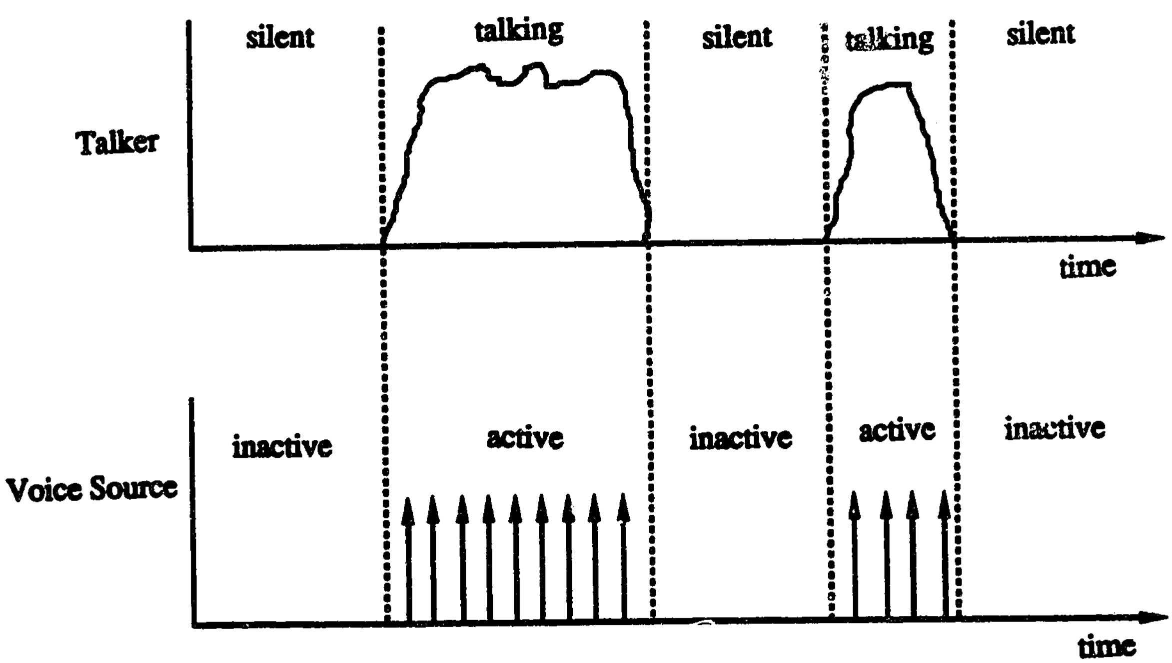


Figure 5.8: Typical voice source behavior

During an active period (talkspurt), the voice source generates packets of fixed length at regular intervals (e.g., synchronous); see the arrows in Figure 5.8. During the *inactive* (silence) period, the voice source does not generate any packets. Actual performance measurements on voice [85] have shown that, on the average, the length of the talkspurts represent about 40% of connect time. Logically this is reasonable, since a voice call is usually a two way conversation. Thus, each caller will be talking for half the time. The other 10% is attributed to pauses between utterances and think times during talking.

Time Assignment Speech Interpolation (TASI)[20] [86, 87] is a technique in which the idle time between talkspurts and pauses during calls are used to accommodate additional calls. This gives rise to what is called *freeze out*. Freeze out results from a voice call trying to access a channel when there are none available, and results in *clipping* the initial portion of the talkspurts. Freeze out should be kept at a minimum to preserve the speech quality.

Voice traffic is different from data traffic in the following aspects [51, 52]:

(1) During a talkspurt, voice packets are produced periodically.

(2) Voice packets are usually of a constant length.

(3) Voice packets cannot experience long queueing or transmission delays (greater than 300ms). Voice packets not transmitted before their deadline are considered lost (which is an aspect of real-time traffic).

20. Digital Speech Interpolation (DSI) is the digital version of TASI, and is used with digitized voice.

(4) Up to 2% of voice packets can be lost without degrading the quality of delivered speech.

(5) For voice users, performance is subjectively perceived, while for data users, performance is objectively perceived. That is, the quality of received voice packets is more important to the end user than a 100% correctness.

Alternative integration techniques:

Three alternatives for voice/data integration exist: circuit switching, packet switching and hybrid switching.

In circuit switching, an end-to-end physical circuit (in the form of time division multiplexing (TDM)) is established. This might be suitable for voice traffic, as the channel would be used by a talker for 40% of the time. For data transmission, TDM is not efficient, and may result in bandwidth wastage. Using TADI[21] (Time Assigned Data Interpolation) techniques, data could be transmitted in the unused voice-assigned channel capacity.

In packet switching, voice and data packets are transmitted independently. An implication of this is that voice packets may arrive at the receiving end out of sequence, and therefore, must be ordered. Packet switching is suitable for bursty data traffic, and may be used for voice traffic if the channel is known to be highly reliable and congestion free.

Hybrid switching [88] combines the advantages of both circuit and packet switching by offering both modes of operation. There are several variations of this approach. In one

21. Similar to TASI, but for data.

variation, voice is circuit switched while data is packet switched. Another variation is to circuit switch bulk data and packet switch voice packets and bursty data packets. Yet, another alternative is to use TADI in circuit switched operation to make better use of channel capacity. The study showed that, at high offered loads, circuit switched voice should be used. Under low offered loads, however, packet switching for voice and data should be employed.

<u>Modeling of Integrated voice/data transmission:</u>

The first step towards modeling voice transmission, is to study the voice packet generation process at the voice source. Voice traffic variations arise from call initiation and termination, and from talkspurts/silence alternation of speakers. A convenient statistical model for voice traffic addressing both of the above requirements is a Markovian model [89-91]. In such a model, it is assumed that the interarrival time of voice calls and their duration, as well as the talkspurt duration and the silence period duration are all exponential random variables. Brady [89], experimentally found that the length of the talkspurt fits well the exponential distribution. However, the distribution of the silence periods was found to be less well approximated by the exponential distribution.

The difficulty in modeling voice traffic arrival in packet switched voice transmission stems from the fact that during the talkspurt, packets arrive periodically. That is, individual voice packet arrivals cannot be modeled using the usually employed Poisson arrival model. One solution to this problem is to approximate the arrival process of voice packets by a Poisson arrival process with batch packets [92]. The batch size is geometric and corresponds to the number of packets generated in a talkspurt, while the interarrival time between packet batches is exponential and corresponds to the length of the silence period. The problem with this model is that it assumes the arrival of batch packets before they would actually arrive in true voice sources. If, however, the packet delay is greater than the packetization time, such an approximation would be more accurate.

The complexity of a two-class CASLANs performance model was discussed in section 5.3. It should be obvious to the reader that an integrated voice and data CASLANs model would be at least as complex. Moreover, modeling voice traffic would make matters even more involved. Therefore, simulation techniques are used in studying integrated CASLANs. The simulation model used is described next.

5.5.2. Simulation Model

In our simulation model, two classes of users exist: voice and data users. Both classes are permitted to transmit their packets, freely, according to the CASLANs protocol. We distinguish two parts to the simulation model: a *network* and a *traffic* part.

The network part of the simulation model behaves exactly as the CASLANs protocol under the following conditions:

(1) All transmitting nodes are equipped with single buffers. That is, once a packet occupies the buffer, no new packets are generated at a node unless the packet currently in the buffer has been successfully transmitted.

(2) No packet loss due to network failure or buffer overflow at the receiving end is

considered.

(3) The transmission medium is noiseless and error free.

(4) For either class, the retry time is the same for all nodes belonging to that class.

For the traffic part of the simulation model, the following assumptions are made:

(1) Identical and independent packet generation process at all data nodes. The packet generation process is exponentially distributed with rate λ_d.

(2) Data packets are of variable length with mean $\bar{T}_d$ for all nodes. The packet length distribution is chosen to be exponential.

(3) Voice sources are constantly active in conversations throughout the entire simulation period. This assumption is known in the literature as the *quasi-static assumption*. In other words, voice traffic is modeled as having alternating talkspurts and silenceperiods.

(4) During a talkspurt, a voice source periodically generates constant length voice packets. The length of the period, L_p, is equal to the time required to accumulate enough voice samples to make a packet. Let S_v represent the voice coding rate (bits/sec) and T_v represent the length of voice packets. L_p is then given by

$$L_p = \frac{T_v}{S_v}$$

The performance measures are the average packet delay for data users and the probability of packet loss for voice users. Because of the single buffer assumption, a voice packet generated at a source with a busy buffer is considered lost[22]. To prevent such a loss, the packet generation period, L_p, is used to represent the laxity for voice packets. Hence, a packet not successfully transmitted during L_p is considered lost and is removed from the buffer, thereby, allowing newly generated voice packets to be admitted to the buffer. For example, if the length of voice packets is 480 (bits) and the voice sampling rate is 64 Kbits/sec, the packet laxity would then be 7500 μs rather than the normal 300 ms defined earlier. It will be shown next that even with such lower laxity, CASLANs perform extremely well, under integrated voice and data traffic.

5.5.3 Numerical Results

In this section, some numerical results obtained from the simulation model above are shown. Effects of retry time values, number of voice and data nodes and data packet arrival rate on the performance of integrated voice and data CASLANs are studied.

In setting the parameters for the simulation runs, the following facts and conditions are considered:

(1) Voice is the real-time traffic in the network, and, therefore, voice packets should be assigned higher priority.

(2) Since voice packets are of fixed length, one can make use of the superiority of factor retry time values by assigning voice sources a retry time that is a factor of the packet length.

22. Because of the periodic nature of voice traffic, blocking loss and channel access loss cannot be treated independently.

(3) Lower priority data packets could be transmitted with a retry time that is a non-factor of the voice packet length.

(4) Priority for voice packets should not be achieved at the expense of excessive data packet delay.

(5) At most 2% of voice packets can be lost while keeping good speech quality.

Let N_d, λ_d, τ_d, and $\bar{T}_d$ be the number of nodes, packet arrival rate per bit time, retry time and mean packet length for data users. We set $N_d = 100$, $\bar{T}_d = 480$ bits, $\tau_d = 180$ (a non-factor retry time value), and vary λ_d. Also, let N_v, τ_v, and T_v be the number of voice sources, the retry time used by voice sources and the voice packet length, respectively. T_v is set[23] to 480 (bits) and the effects of varying the retry time and the number of simultaneous voice users[24] are studied. The average length of talkspurts and silence periods are taken to be 1.2 and 1.8 seconds respectively [85].

Under light to medium data offered load, 1000 voice sources could be accommodated in an integrated CASLAN, with the above set parameters, with no packet loss observed. Therefore, such a case is not pursued any further.

Tables 5.2-5.5, show performance results of integrated voice and data CASLANs, under high to heavy offered load conditions, at different values of τ_v with $N_v = 100, 200, 500$ and 1000 sources respectively. From the results shown in these tables, the following observations are made:

23. The choice of $\bar{T}_d$ and T_v to be 480 bits is simply done for consistency with the rest of the thesis.

24. Since voice users are constantly involved in voice calls, the number of simultaneous voice users would be N_v. Up to 1000 simultaneous users are allowed.

τ_v	$N_d = 100\ T_d = 480$ bits ; $N_v = 100\ T_v = 480$; Bit rate =50 Mbps					
	Percentage of voice packets lost			Average delay for data packets(μs)		
	$\lambda_d = 0.0001$	$\lambda_d = 0.001$	$\lambda_d = 0.01$	$\lambda_d = 0.0001$	$\lambda_d = 0.001$	$\lambda_d = 0.01$
100	0.0	0.0	0.0	818.7	972.4	1014.6
120	0.0	0.0	0.0	821.3	976.1	1012.7
140	0.0	0.0	0.0	821.9	975.5	1018.3
160	0.0	0.0	0.0	825.9	976.6	1021
180	0.62	1.62	2.25	823.9	986.7	1016.5
200	0.0	0.17	0.37	824	975.2	1015.8
220	0.0	0.0	0.0	825.5	974.9	1008.4
240	0.0	0.0	0.0	824.8	977.8	1013.1
260	0.0	0.0	0.51	819.5	975.9	1014.7
280	0.0	0.0	0.36	815.5	977.3	1015.2

Table 5.2: Performance results of an integrated voice and data CASLAN (with 100 voice sources)

τ_v	$N_d = 100\ T_d = 480$ bits ; $N_v = 200\ T_v = 480$; Bit rate =50 Mbps					
	Percentage of voice packets lost			Average delay for data packets (μs)		
	$\lambda_d = 0.0001$	$\lambda_d = 0.001$	$\lambda_d = 0.01$	$\lambda_d = 0.0001$	$\lambda_d = 0.001$	$\lambda_d = 0.01$
100	1.46	2.4	2.1	844.5	1004.8	1035.7
120	0.0	0.0	0.0	851.8	1016.8	1033.2
140	0.0	0.0	0.0	838.5	1017.9	1035.2
160	0.0	0.0	0.0	839.4	1026.3	1033.3
180	1.64	2.98	3.7	842.2	1035.0	1038.7
200	0.4	3.1	3.65	836.2	1008.0	1026.7
220	0.0	0.0	0.0	834.9	1016.0	1047.7
240	0.0	0.0	0.0	843.0	1030.5	1035.4
260	0.0	0.6	1.9	839.9	1018.2	1049.6
280	0.0	0.0	0.79	834.6	1025.4	1038.5

Table 5.3: Performance results of an integrated voice and data CASLAN (with 200 voice sources)

τ_v	$N_d = 100$ $T_d = 480$ bits ; $N_v = 500$ $T_v = 480$; Bit rate =50 Mbps					
	Percentage of voice packets lost			Average delay for data packets (μs)		
	$\lambda_d = 0.0001$	$\lambda_d = 0.001$	$\lambda_d = 0.01$	$\lambda_d = 0.0001$	$\lambda_d = 0.001$	$\lambda_d = 0.01$
100	6.54	7.41	9.0	871.9	1017.4	1044.8
120	0.0	0.0	0.0	898.7	1076.2	1111.4
140	0.0	0.0	0.0	911.5	1091.5	1096.0
160	0.0	0.0	0.0	912.4	1066.2	1099.7
180	2.97	6.31	5.9	911.2	1100.4	1080.5
200	10.4	15.3	16.5	860.5	1030.0	1043.9
220	0.0	1.2	1.8	895.0	1101.7	1099.6
240	0.0	0.41	0.85	899.6	1070.4	1100.8
260	0.6	0.94	4.9	902.7	1094.7	1095.4
280	2.3	3.5	4.7	909.1	1079.6	1103.6

Table 5.4: Performance results of an integrated voice and data CASLAN
(with 500 voice sources)

τ_v	$N_d = 100$ $T_d = 480$ bits ; $N_v = 1000$ $T_v = 480$; Bit rate =50 Mbps					
	Percentage of voice packets lost			Average delay for data packets (μs)		
	$\lambda_d = 0.0001$	$\lambda_d = 0.001$	$\lambda_d = 0.01$	$\lambda_d = 0.0001$	$\lambda_d = 0.001$	$\lambda_d = 0.01$
100	7.8	12.8	13.9	883.4	1028.0	1169.8
120	0.0	0.0	0.0	1051.9	1209.4	1232.5
140	1.7	2.7	3.1	969.2	1201.7	1219.4
160	1.3	1.6	1.9	1052.3	1208.2	1226.3
180	8.7	9.8	10.3	1010.5	1183.4	1195.7
200	12.6	14.5	15.1	869.7	1038.8	1164.1
220	2.9	4.6	5.2	1017.7	1188.6	1220.4
240	1.4	2.3	2.7	1049.6	1204.3	1219.5
260	2.8	5.8	6.9	1039.4	1195.5	1203.6
280	1.6	5.2	5.7	1024.6	1189.2	1201.2

Table 5.5: Performance results of an integrated voice and data CASLAN
(with 1000 voice sources)

(1) Except for a few odd values, the performance of CASLANs in an integrated voice and data environment is excellent. For instance, at N_v = 100, Table 5.2, the probability of voice packet loss is almost always below the 2% mark. Even with 1000 simultaneous voice sources, Table 5.5, one can set the parameters such that no packet loss is encountered, e. g. τ_v = 120 bits.

(2) Again, factor retry time values have shown their superiority. For instance, only at τ_v = 120 and 160 (bits) (factor retry time values), is the probability of voice packet loss always kept below the 2% mark.

(3) The superiority of factor retry time values is more evident at higher values of N_v. The reason being that as N_v increases so does the hub utilization by voice sources (using factor retry time values) and subsequently hub hogging (see section 5.2) by voice sources.

(4) The average delay for data packets is still very reasonable despite the large numbers of voice and data users and the high offered load. For example, with N_v = 500, N_d = 100 and a data packet arrival rate of 0.01 per bit time per node, the average delay of data packets is still below 1.2 ms.

Indeed, the results in Tables 5.2-5.5, show that CASLANs are very good candidates for voice and data integration.

5.6 Summary

This chapter studied and analyzed the performance and behavior of prioritized

CASLANs. It was shown that by simply varying retry time values, priority could be effectively implemented on CASLANs.

Behavioral analysis of two-class CASLAN showed that assigning a retry time that a factor of the packet length provides higher priority over any greater retry time value, and some lower retry time values subject to some, load dependent, range. It was also shown that a priority class, using such a retry time value, seems to hog the hub, thus nullifying, to some extent, the effect of existence of packets belonging to a lower priority class (not using a factor retry time value), and, indeed, resulting in an effective priority scheme.

A two-class performance model that involves exact analysis of tagged users was considered. Such a model was shown to be very involved and computationally expensive. Therefore, another, less involved but more simplified, model was used to evaluate the performance of prioritized CASLANs. The model assumes independence between inter-packet arrival times. Such a model, however, proved to be accurate only for moderately loaded CASLANs.

Analysis of integrated voice and data CASLANs has shown that CASLANs are very good candidates for integrated voice and data applications. Indeed, it was shown that by proper choice of network parameters (e.g., a retry time that is a factor of the packet length) voice packet loss could be eliminated, even in large systems (1000 voice sources) and high data offered loads (100 data nodes with an arrival rate of 0.01 packets per bit time per node).

Chapter 6

Conclusions and Future Work

6.1. Summary

This thesis has dealt with the modeling and analysis of Collision-Avoidance Star Local Area Networks (CASLANs). Research on CASLANs took three different directions: behavioral and performance analysis, analysis of CASLANs in a time-constrained environment and a study of prioritized CASLANs.

Behavioral analysis of CASLANs has shown that:

(1) Increasing the retry time does not necessarily mean an increase in the average delay. Indeed, the average delay at a retry time that is a factor of the packet length is lower than that at every greater retry time values, and some lower retry time values subject to some, load dependent, range.

(2) A packet transmitted in a CASLAN, in which the retry time is a factor of the packet length, is guaranteed access to the channel.

(3) Under heavy load conditions, a symmetric CASLAN, with a retry time that is a factor of the packet length, has a bounded delay.

Previous CASLANs performance models [39-45] have shown the average packet delay to be always increasing, with increasing retry time, hence, not reflecting the findings above. In chapter 3, a new CASLANs performance model was, therefore, introduced and analyzed. The model follows a tagged user in an exact analysis, and the rest of the users' behavior was approximated by mimicking that of the tagged user at steady state, using an iterative approach. CASLANs were modeled as a polling system where the hub chooses at random one of the idle, ready or tagged users and start serving it. Performance results have shown that the model accurately predicts the performance of CASLANs, and is more accurate than any previously devised performance model. Such superior performance is attributed to the fact that the model does not assume independence between packet retry instants at the hub.

The model in chapter 3 was extended to accommodate hard real-time traffic with exponentially distributed packet laxities. The model was shown to be very accurate. It is interesting to note that even though these models were devised for CASLANs, they could be extended to model the performance of a wide variety of local area networks protocols.

By analyzing the performance of CASLANs for both soft and hard real-time applications, the following was shown:

(1) In general, CASLANs perform very well under real-time constraints. Indeed, CASLANs seem to experience very low loss probabilities even under high offered loads.

(2) Factor retry time values seem to possess local superiority in terms of a lower packet loss probability. Therefore, indicating that superior performance of factor retry time values extends beyond average performance results to delay distributions.

Analysis of prioritized CASLANs has revealed the following:

(1) A retry time that is a factor of the packet length provides higher priority over any greater retry time, and some lower retry time values, that fall within its range.

(2) In an integrated voice and data environment, CASLANs were shown to accommodate a large number of voice sources without experiencing considerable voice packets loss. In fact, the network parameters could be set such that no packet loss is encountered, even under extremely high data offered load conditions.

A two-class prioritized CASLANs model was analyzed. The model represents CASLANs as a polling system where the hub chooses at random one of the users of either class. Since it is an extremely difficult and mathematically involved process to keep track of interpacket arrival times of packets from two tagged users at the hub, the model dropped such a requirement. Therefore, the model was not capable of accurately capturing the relations between the packet transmission times and the retry times. Since such a relation is evident at high to heavy offered loads, the model is intended for moderately loaded prioritized CASLANs, or for variable packet length environments.

6.2 Future Work

In chapter 2, the performance of retry time values that are factors of the packet length was extensively studied. Several new performance and behavioral characteristics of CASLANs employing such retry time values were shown. Specifically, the guaranteed packet delivery and the bounded delay, at heavy load. These results, however, are confined to fixed-packet-length CASLANs.

In many practical situations, however, this is not the case. That is, packets are of variable length. It might be worthwhile, then, to impose a restriction on packets inserted onto the channel such that they appear to be multiples of the retry time. In effect, before packets are transmitted onto the channel, they are padded with enough dummy bits to force the packet length to become a multiple of the retry time. This technique would then guarantee successful transmission for all generated variable-length packets.

In modeling local area networks in general, and in this thesis in particular, the single buffer assumption has been widely used. The case where the number of buffers is finite has been avoided because of its involvement and mathematical intractability. Consideration of such a case, however, might be essential especially in modeling real-time systems. This is because in many real-time applications multiple-packet message laxities are used, rather than each packet having its own laxity.

In modeling hard real-time CASLANs, exponentially distributed packet laxities were used. Such a distribution, however, may not be representative of packet laxities in many real-time applications. A model employing other distributions for packet laxities might be, then, worthwhile considering. On such distribution is the Coxian distribution, due to Cox [93]. In [94], it is argued that the Coxian distribution can be used to approximate very closely any general probability distribution function. It was also shown in [94] that the Coxian distribution can be represented by a number of exponentially distributed stages, where a job can exit the system with a certain probability at any of these stages. It is then feasible to use the Coxian distribution for packet laxities, where the packet loss at each stage is exponentially distributed. This, however, would come at the expense of an increased state space by a factor equivalent to the number of stages.

6.3 Concluding Remarks

From the author's point of view, the work in this thesis has demonstrated the following performance and behavioral attributes of CASLANs:

(1) Regardless of the application sought to be implemented on fixed-packet-length CASLANs, an increased propagation delay (or retry time) between a node and the hub may result in improved performance results. Such improved performance was observed at every retry time that is a factor of the packet length.

(2) Use of factor retry time values results in guaranteed packet transmission and bounded delay (at heavy load).

(3) CASLANs are very good candidates for use in real-time applications and for applications that involve the integration of voice and data.

During the course of this study, several contributions towards modeling and analysis of CASLANs were made:

(1) An exact, heavy load, performance model of CASLANs was introduced. The model was used to show that the delay of CASLANs, under heavy load conditions, is bounded, and that packet transmission order is round-robin.

(2) A new performance model for CASLANs has been introduced. The model was shown to be more accurate than any previously devised CASLANs performance model.

(3) A hard real-time performance model of CASLANs has been presented. In this model packet laxities were chosen to be exponentially distributed to reduce the complexity of the model.

(4) A prioritized CASLANs performance model has been introduced for moderately loaded systems.

References:

[1] V. Li, "Multiple Access Communications Networks," *IEEE Communications Mag.*, Vol. 25, No. 6, June 1987, pp. 41-48.

[2] S. R. Sachs, "Alternative Local Area Network Access Protocols," *IEEE Communications Magazine*, Vol. 26, No.3, Mar. 1988, pp. 25-45.

[3] J. Kurose, M. Schwartz and Y. Yemini, "Multiple-Access Protocols and Time-Constrained Communication," *ACM Computing Surveys*, Vol.16, No.1, Mar. 1984.

[4] F. Tobagi, " Multiaccess Protocols in Packet Communication Systems," *IEEE Trans. on Comm.*, Vol. com-28, No. 4, April 1980, pp. 468-488.

[5] IEEE Standard 802.5. "Token-Passing Ring Access Method," Dec. 13, 1984.

[6] IEEE Standard 802.4. "Token-Passing Bus Access Method," Dec. 17, 1984.

[7] I. Chlamtac, W. Franta, and K. D. Levin, "BRAM: the Broadcast Recognizing Access Method," *IEEE Trans. Commun.*, Vol. com-27, Aug. 1979, pp. 1183-1190.

[8] L. Kleinrock and M. Scholl, "Packet Switching in Radio Channels: New Conflict-Free Multiple Access Schemes," *IEEE Trans. Commun.*, Vol. com-28, July 1980.

[9] F. Borgonovo, L. Fratta, F.Tarini, and P. Zini, "L-Expressnet: a Communication Protocol for Local Area Networks," in *Proc. IEEE INFOCOM*, San Diego, CA, Apr. 1983.

[10] L. F. Fratta, F. Borgonovo, and F. A. Tobagi, "The Expressnet: A Local Area Communication Network Integrating Voice and Data," in *Proc. Int. Conf. on Performance of Data Communications Systems and* Their Applications, Paris, France, Sept. 1981.

[11] C. W. Tseng and B.U. Chen, "D-net, A New Scheme for High Data Rate Optical Local Area Networks," *IEEE J. Select. Areas Commun.*, Vol. SAC-1, No.3, Apr. 1983.

[12] J. O. Limb and C. Flores, "Description of Fasnet-a Unidirectional Local Area Communications Network," *Bell Syst. Tech. J.*, Vol. 61, No. 7, Sept. 1982, pp.

1413-40.

[13] M. Gerla, P. Rodrigues, and C. Yeh, "U-net:a Unidirectional Fiber Bus Network," in *Proc. FOC/LAN 84*, 1984.

[14] P. A. Rodrigues, L. Fratta, and M. Gerla, "Tokenless Protocol for Fiber Optics Local Networks," in *ICC Conf. Proc.*, Amsterdam, May 1984.

[15] N. Abramson, "The ALOHA System - Another Alternative for Computer Communications," *Proc. of the AFIPS Fall Joint Computer Conference*, Nov. 1970.

[16] L. Kleinrock and F. A. Tobagi, " Packet Switching in Radio Channels: Part I-Carrier Sense Multiple Access Models and their throughput-delay Characteristics," *IEEE Trans. Comm.*, Vol. com 23, No. 12, pp. 1400-1416.

[17] R. M. Metcalfe and D. R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks," *Comm. Ass. Comp. Mach.*, Vol. 19, July 1976, pp. 395-404.

[18] M. L. Molle and L. Kleinrock, "Virtual Time CSMA: Why Two Clocks Are Better Than One," *IEEE Trans. Commun.*, Vol. Com-33, No. 9, Sept. 1985.

[19] L. Kleinrock and Y. Yemini, "An Optimal Adaptive Scheme for Multiple Access Broadcast Communication," in *ICC Conf. Proc.*, Chicago, Ill., June 1977.

[20] J. Capetanakis, "Tree Algorithms for Packet Broadcast Channels," *IEEE Trans. Info. Theory*, Vol. 17.25, Sept. 1979, pp.505-515.

[21] J. F. Hayes, "An Adaptive Technique for Local Distribution," *IEEE Trans. Commun.*, Vol. com-26, Aug. 1978, pp. 1178-1186.

[22] J. Massey, "Collision-Resolution Algorithms and Random-Access communication," in *Multi-user Comm., G. Lomgo, ed.*, New York: springer-verlag 1981, pp. 73-137.

[23] B. Tsybakov and V. Mikhailov, "Slotted Multiaccess Packet-Broadcasting Feedback Channel," *Problems in Information Transmission*, Vol.14, Oct. 1978, pp. 32-59.

[24] A. Greenberg, P. Flajolet and R. Lacher, "Estimating the Multiplicities of Conflicts to Speed Their Resolution in Multiple Access Channels," *J. ACM.*, Vol. 74, No. 2, Apr. 1987, pp. 289-325.

[25] I. Cidon and M. Sidi, "Conflict Multiplicity Estimation and Batch Resolution Algorithms," *IEEE Trans. Info. Theory*, Vol. IT-34, No.1, Jan. 1988, pp. 101-110.

[26] R. Gallager, "Conflict Resolution in Random Access Broadcast Networks," in *Proc. AFOSR Workshop Comm. Theory Appl.*, Sept. 1978, pp. 74-76.

[27] J. Mossely and P. Humblet, "A Class of Efficient Contention Resolution Algorithms for Multiple Access Channels," *IEEE Trans. Comm.*, Vol. com-33, No. 2, Feb. 1985, pp. 145-151.

[28] S. Panwar, D. Towsely, and J. Wolf, "On the Throughput of Degenerate Intersection and First-Come First-Serve Collision Resolution Algorithms." *IEEE Trans. Info. Theory*, Vol. IT-31, No.2, Mar. 1985, pp. 274-279.

[29] W. M. Kiesel and P. J. Kuehn, "A New CSMA/CD Protocol for Local Area Networks with Dynamic Priorities and Low Collision Probability," *IEEE J. Selected Areas Commun.*, Vol. SAC-1, No. 5, Nov. 1983.

[30] A. Takagi, S. Yamada, and S. Sugawara, "CSMA/CD with Deterministic Contention Resolution," *IEEE J. Selected Areas in Commun.*, Vol. SAC-1, No. 5, Nov. 1983.

[31] M. E. Ulug, "A Fiber Optic Contension Bus with Bounded Delays," *IEEE Jour. Selec. Areas in Comm.*, Vol. SAC-3, No. 6, Nov. 1985, pp. 908-915.

[32] M. Gerla, P. Rodrigues, and C. Yeh, "Buzz-net:A Hybrid Random Access/Virtual Token Local Network," in *Proc. Globecom '83*, S. Diego, CA, Dec. 1983.

[33] B. Abeysundara and A. E. Kamal, "Z-net: A Dual Bus Fiber-Optic LAN using Active and Passive Switches," *in Proc of IEEE INFOCOM*, Apr. 1989, pp. 19-27.

[34] A. Kamal and B. Abeysundara, "X-net: A Dual Bus Fiber-Optic LAN Using Active Switches," *submitted ACM SIGCOM*, 1989.

[35] M. Rios and N. Georganas, "A Hybrid Multiple Access Protocol for Data and Voice-Packet Over Local Area Networks," *IEEE Trans. Comput.*, Vol. C-34, Jan. 1985, pp. 90-94.

[36] A. Mok and S. Ward, "Distributed Broadcast Channel Access," *Computer Networks*, 1979, pp. 327-335.

[37] E. S. Lee and P. I. P. Boulton, "The Principles and Performance of HUBNET: A 50 Mb/s Fiber Optics Local Area Network," *IEEE Jour. Selec. Areas Communications,* Vol. SAC-1, Nov. 1983, pp. 711-720.

[38] F. Closs and R. P. Lee, "A Multi-Star Broadcast Network for Local Area Communications Networks," *in: Local Networks for Computer Communications,* A. west and P. Jansen (eds.), North Holland, 1981, pp. 61-80.

[39] A. Albanese, "Star Networks with Collision Avoidance," *Bell Sys. Tech. Jour.,* Vol. 62, No.3, Mar. 1983, pp.631-638.

[40] T. Suda, Y. Yemini and M. Schwartz, "Tree Networks with Collision Avoidance Switches," *in Proc. of IEEE INFOCOM* 1984, pp. 105-113.

[41] V. C. Hamacher and W. M. Loucks, "Packet Transfer Delays in Local Area Computer Networks," *INFOR,* Vol. 23, No. 3, Aug. 1985, pp. 248-274.

[42] T. Suda and K. Goto, "Performance Study of a Tree LAN with Collision Avoidance," *in Proc. of IEEE INFOCOM,* Apr. 1989, pp 59-68.

[43] G.K. Janssens, "Dependent Collision Performance Model for a Star Topology Local Area Network," Tech. rep. No. , Dept of Comp. Sci., State Univ. of Antwerp.

[44] A. E. Kamal and V. C. Hamacher, "Analysis of a Star Local Area Network with Collision Avoidance," *in Proc. of IEEE INFOCOM* 1986, pp. 546-555.

[45] A. E. Kamal, "A Performance Model for a Star Network," *in Proc. of IEEE GLOBECOM* 1986, pp. 12-18.

[46] G.K. Janssens, "A Performance Model for a Conflict-Free Access Method of a Tree-Structured Local Area Network," *Proc. Euro. Simulation Multicon,* July 7-10, 1987, pp. 23-26.

[47] P.I.P. Boulton, R.E. Soper and E.S. Lee, "Simulation of Two LANs," Univ. of Toronto Tech Report No. CSRI-233, March 1990.

[48] E. S. Lee, P. I. P. Boulton, and B. W. Thomson, "HUBNET Performance Measurements," *IEEE Jour. Selec. Areas Communications,* Vol. SAC-6, No. 6, July 1988, pp. 1025-1032.

[49] F.E. Ross, "FDDI-a Tutorial," *IEEE Communications Magazine,* Vol. 24, No. 5, May 1986, pp. 10-17.

[50] J. A. Stankovic, "Misconceptions About Real-Time Computing: A Serious Problem for next Generation Systems," *IEEE Computer Mag.*, Vol. 21, No. 10, Oct. 1988, pp. 10-19.

[51] M. Malek, "Integrated Voice and Data Communications Overview," *IEEE Communications Mag.*, Vol.26, No. 6, June 1988, pp. 5-15.

[52] R. Pokress, Ed., Special Issue on Integrated Services Digital Networks, *IEEE Common. Mag.*, Jan. 1984.

[53] J. G. Gruber and N. H. Le, "Performance requirements for Integrated Voice/data Networks," *IEEE Journal on selected Areas in commun.*, Vol. SAC-1, No. 6, Dec. 1983, pp. 981-1005.

[54] C. Weinstein and J. Forgie, "Experience with Speech Communication in Packet Networks," *IEEE J. Select. Areas commun.*, Vol. SAC-1, Dec. 1983, pp.963-980.

[55] M. Sloman and S. Prince, "Local Network Architecture for Process Control," in *Local Networks for Computer Communications*, Amsterdam: North-Holland, 1981, pp.407-427.

[56] J. Blazewicz and J. Weglarz, "Scheduling Under Resourse Constraints Achievements and Prospects," in *Performance of Computer Systems*, M. Arato, A Butrimento, and E. Gelenbe, Eds. New York: North-Holland,1979.

[57] E. D. Jensen, C. D. Locke and H. Tokuda, "A Time-Driven Scheduling Model for real-Time Operating systems," *IEEE Real-Time Sys. Symp.*, 1985, pp. 112-122.

[58] W. Zhao, K. Ramamritham and J. Stankovic, "Preemptive Scheduling Under Time and Resource Constraints," *IEEE Trans. on Comp.*, Vol. C-36, No. 8, Aug. 1987, pp. 949-960.

[59] E. Arthurs and B. Stuck, "A Theoritical Traffic Performance Analysis of an Integrated Voice-Data Virtual Circuit Packet Switch," *IEEE Trans. Commun.*, Vol. com-27, July 1979, pp. 1104-1111.

[60] O. Mowafi and W. Kelly, "Integrated Voice/Data Packet Switching Techniques for Future Military Networks," *IEEE Trans. Commun.*, Vol. com-28, Sept. 1980, pp. 1655-1662.

[61] J. Santos, J. Orozco and O. Alimenti, "Performance Evaluation of Standard Lan Protocols in Time-Constrained environments," *in Proc. of IEEE*

INFOCOM, 1989, pp. 1020-1028.

[62] B. Kim, "Two Adaptive Token Ring Strageries for Real-Time Traffic," *in Proc. IEEE Computer Networking Symp.*, N.J., Dec. 1983, pp. 119-121.

[63] K. Ramamrithan, "Channel Characteristics in Local Area Hard Real-Time Systems," *Computer Networks and ISDN Syst.* 1987.

[64] W. Hoffmann and T. Kersting, "Simulation of Ethernet Under Real-Time Conditions," *Process Automation*, 1984, pp. 29-34.

[65] D. Hutchison and M. Merabti, "Ethernet for Real-Time Applications," *IEE Jour.*, 1987.

[66] N. F. Maxemchuk, "A Variation on CSMA/CD That Yields Movable TDM Slots in Integrated Voice/Data Local Networks," *Bell Syst. Tech. J.*, 1982, pp. 1527-1550.

[67] G.J. Nutt and D.L. Bayer, "Performance of CSMA/CD Networks Under Combined Voice and Data Loads," *IEEE Trans.*, Vol. com-30, No. 1, 1982, pp. 6-11.

[68] I. Chlamtac and M. Eisenger, "Voice/Data Integration on Ethernet- Backoff and Priority Considerations," *Computer Communications*, Vol.6, No. 5, 1983, pp. 236-244.

[69] R.K. Goel and A.K. Elhakeem, "A Hybrid FARA/CSMACD Protocol for Voice-Data Integration," *Computer Networks and ISDN syst.*, Vol.9, 1985, pp. 223-240.

[70] C. Lea and J.S. Meditch, "A Channel Protocol for Integrated Voice/Data Applications," *IEEE J.Selec.Areas in Commun.*, Vol. SAC-5, No. 6, July 1987, pp. 939-947.

[71] Z. Gao and K. Vastola, "Performance Analysis of an Integrated Voice/Data Protocol," *in Proc. of ICC*, 1988, pp. 823-828.

[72] C. Yuan and J. Silvester, "An Integrated Voice/Data Protocol for Local Area Bus Networks," *in Proc. of IEEE GLOBECOM*, 1988, pp. 1813-1817.

[73] J. Kurose, M. Schwartz and Y. Yemini, "Controlling Time Window Protocols for Time-Constrained Communication in a Multiple Access Environment," *in Proc. 8th IEEE Int'l Data Comm. Symp.*, Oct. 1983, pp. 75-84.

[74] W. Zhao and K. Ramamritham, "Virtual Time CSMA Protocols for Hard Real-Time Communication," *IEEE Trans. on Soft. Eng.*, Vol. SE-13, No.8, Aug. 1987, pp. 938-952.

[75] D. Barrer, "Queueing Systems with Impatient Customers and Indifferent Clerks," Operations Res., Vol. 5, 1967, pp. 644-649.

[76] D. Barrer, "Queueing Systems with Impatient Customers and Ordered Service," Operations Res., Vol. 5, 1967, pp. 650-656.

[77] F. Baccelli and G. Hebuterne, "On Queues with Impatient Customers," *in Performance '81*, Amsterdam, 1981, pp. 159-179.

[78] B. Gavish and P. Schweitzer, "The Markovian Queue with Bounded Waiting Time," *Management Sci.*, Vol. 23, Aug. 1977, pp. 1349-1357.

[79] F. Baccelli, P. Boyer and G. Hebuterne, "Single-Server Queues with Impatient Customers," *Adv. Appl. Prob.*, Vol. 16, 1984, pp. 887-905.

[80] I. Rubin and M Ouaily, "Performance of Communication and Queueing Processors under Message Delay Limits," *in Proc. IEEE GLOBECOM*, 1988, pp. 501-505.

[81] I. Rubin and M. Ouaily, "Performance Analysis of Message Delay Limited Synchronous Communication and Queueing Systems," *in Proc. IEEE INFOCOM*, 1989, pp. 51-58.

[82] F. A. Tobagi, "Carrier sense Multiple Access with Message-Based Priority Function," *IEEE Trans. Commun.*, Vol. com-30, Jan. 1982.

[83] G. L. Choudhury and S. S. Rappaport, "Priority Access Schemes Using CSMA-CD," *IEEE Trans. Commun.*, Vol. Com-33, July 1985.

[84] I. Gitman and H. Frank, "Economic Analysis of Integrated Voice and Data Networks: A Case Study," *Proc. IEEE*, Vol.66, Nov. 1978, pp. 1549-1570.

[85] P. Brady, "A Statistical Analysis on On-Off Patterns in 16 Conversations," *Bell Syst. Tech.J.*, Vol. 47, No. 1, Jan. 1968, pp. 73-91.

[86] S. Campanella, "Digital Speech Interpolation," *COMSAT Tech. Rev.*, Spring 1976.

[87] S. Campanella, "Digital Speech Interpolation Techniques," Nat. Telecommun. Conf., Dec. 1978.

[88] M. J. Ross and O. A. Mowafi, "Performance Analysis of Hybrid Switching Concepts for Integrated Voice/Data Communications," *IEEE Trans. Commun.*, Vol. com-30, No.5, May 1982, pp. 1073-1087.

[89] P. T. Brady, "A Model for Generating On-Off Patterns in Two-Way Conversations," *Bell Syst. Tech. J.*, Vol. 48, Sept. 1969.

[90] D. Minoli, "Issues in Packet Voice Communication," *Proc. IEE*, Vol. 126, No. 8, Aug. 1979, pp. 729-740.

[91] T. Bially, A. Mclaughlin and C. Weinstein, "Voice Communication in Integrated Digital Voice and Data Networks," *IEEE Trans. on Commun.*, Vol. com-28, No. 9, Sept. 1980, pp. 1478-1490.

[92] D. Karvelas and A. Leon-Garcia, "Performance of Integrated Packet Voice/Data Token-Passing Rings," *IEEE Journal on Selected Areas in Communications*, Vol. SAC-4, No. 6, Sept. 1986, pp. 823-832.

[93] D.R. Cox, "A Use of Complex Probabilities in the Theory of Stochastic Processes," *Proc., Cambridge Phil. Soc.*, Vol. 51, 1955, pp. 313-319.

[94] E. Gelenbe and I. Mitrani, *Analysis and Synthesis of Computer Systems*, Academic Press, London, 1980.

[95] H. S. Hassanein and A. E. Kamal, "A Study of the Behavior of Hubnet," *Submitted for publication.* Also, available as technical report TR90-1, Department of of Computing Science, University of Alberta, Edmonton, Alberta, Canada, Jan. 1990.

[96] H. S. Hassanein and A. E. Kamal, "A New Guarenteed-Packet-Delivery Protocol for Collision-Avoidance Star Local Area Networks," *Can. Conf. on Elec. & Comp. Eng.*, Ottawa, Canada, Sept. 4-6, 1990, 4pp..