ABSTRACT

INVESTIGATION OF THE PL CEBUS PERFORMANCE WITH AND WITHOUT ACKNOWLEDGMENT

by
Menghan Pan

The power line implementation of the consumer electronic bus (PL CEBUS) is a promising and inexpensive approach for home automation. Since the introduction of the PL CEBUS standards, there has been increasing efforts on evaluating its performance. However, all the works have been performed for unacknowledged networks. This thesis presents the first successful evaluation of the PL CEBUS with acknowledment. Three different cases, namely, 600, 300 and 100 bits packet sizes, have been considered. The evaluation included the simulation of performance parameters such as message and packet delays, message throughput, and channel throughput. Acknowledged network performance has been confirmed to function well in terms of the delays and message throughputs over the practical range of the normalized offered load. For larger load region, the acknowledged PL CEBUS provides a more reliable performance, but at the expense of increased delays and reduced throughputs when compared with the unacknowledged PL CEBUS.

INVESTIGATION OF THE PL CEBUS PERFORMANCE WITH AND WITHOUT ACKNOWLEDGMENT

by
Menghan Pan

A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Electrical Engineering

Department of Electrical and Computer Engineering

October 1993

APPROVAL PAGE

Investigation of the PL CEBUS Performance With and Without Acknowledgment

Menghan Pan

Dr. Constantine N. Manikopoulos, Thesis Advisor (date)
Associate Professor of Electrical and Computer Engineering
New Jersey Institute of Technology

Dr. Mengchu Zhou, Committee Member (date)
Assistant Professor of Electrical and Computer Engineering
New Jersey Institute of Technology

Dr. George E. Antoniou, Committee Member (date)
Visiting Professor of Electrical and Computer Engineering
New Jersey Institute of Technology

BIOGRAPHICAL SKETCH

Author: Menghan Pan

Degree: Master of Science in Electrical Engineering

Date: October, 1993

Author: Menghan Pan

Degree: Master of Science in Electrical Engineering

Date: October, 1993

Undergraduate and Graduate Education:

- Master of Science in Electrical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 1993
- Bachelor of Science in Electrical Engineering,
  University of Electronic Science and Technology of China,
  Chengdu, China, 1984

Major: Electrical Engineering

Work Experience:

- 1990-1991: Manager, AT&T China, Inc., Beijing, China.
- 1989-1990: System Engineer, AT&T China, Inc., Beijing, China.
- 1984-1989: System Engineer, Xinhua News Agency, Beijing, China.

This thesis is dedicated to
my dearest husband
Jian

ACKNOWLEDGMENT

The author gratefully acknowledges the encourangement and guidance offered by her advisor Professor C. N. Manikopoulos. The author would also like to thank the thesis committee members, Professor M. Zhou and Visiting Professor G. E. Antoniou for taking time to review this work.

The use of the LANSF protocol developed by Drs. Pawel Gburzynski and Piotr Rudnicki of the University of Alberta is acknowledged. The help offered by Dr. David Zhang is also acknowledged.

The author thanks her mother and father, mother-in-law and father-in-law for their love, support, and encouragement. The continuing assitance and help of her mother-in-law make it possible to finish this thesis in time.

Deepest thanks go to my dear husband for his love, understanding and help in numerous ways without which this thesis would simply be impossible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The Consumer Electronics Group of the Electronic Industries Association (EIA) initiated an effort in standardizing home communication networks for consumer products in 1984[1]. The result of the effort is the release of the Consumer Electronic Bus (CEBUS) in 1989[2], which is a local area network (LAN) for communication and control within a house. A revision to the first draft of CEBUS was then made and released in 1992[3]. It is now a well accepted standard with standardized communication interface to six different physical communication media [1-6] including the Power Line Bus (PLBUS), the Twisted-Pair Bus (TPBUS), the Coaxial Bus (CXBUS), the Infrared Bus or Single-Room Bus (SRBUS), the Radio Frequency Bus (RFBUS), and the Fiber-Optic Bus (FOBUS). Out of the six communication media, the Power Line CEBUS network appears to be the easiest and the most inexpensive to install because the CEBUS uses the 60 Hz power line as the main retrofit medium which is available in almost every house/office building.

The CEBUS is intended to support home communication for home appliances, entertainment facilities, lighting automation, security monitoring and control among many others. It is based on a packet message format which comprises the control and information fields. For implementation simplicity the CEBUS uses Carrier Sense Multiple Access (CSMA) protocol with contention detection (CSMA/CD) for channel access. With CSMA, the medium is sensed by a node for activity and if no activity is detected in the channel, it then transmits its packet. If contention is detected, the node aborts its transmission and attempts to access the channel again. In other words, if two stations sense the channel to be idle and begin transmitting simultaneously they will both detect the collision almost immediately and abruptly stop transmitting as soon as the collision is detected. The advantage of this access method is that the only information required by the transmitting node is the state of the medium. One of the main concerns with CSMA protocol is that the frequency of collision increases as the offered load rises due to more simultaneous transmission attempts. This results in decreased throughput with increased load [7,8] since collision requires all active nodes to cease transmission and backoff. By using the channel bandwidth for backoff, which otherwise would have been used for data transfer, the throughput diminishes. CSMA utilizes the round robin queueing scheme to provide equal opportunity to transmit within a priority. Three priority classes of messages, namely, HIGH, STANDARD, and DEFERRED are supported. The detailed design will be discussed in Chapter 2.

Other channel access schemes reported in the literature include the Broadcast Recognizing Access Method (BRAM) investigated by Chlamtac et. al.[9], the Modified BRAM (MBRAM) by Signorle et. al[10], the Modified CSMA by Bertan, and the generalized CSMA/CD by Kiesel et. al. [11].

The BRAM is a random access protocol, which is applicable to the network where channel access is not regulated by a single node, while each node is delaying its attempt to seize the channel. Hence the delay time is proportional to the difference between the index of the node accessing the channel and the index of the node last transmitting. The BRAM protocol guarantees a collision free medium with fair access to all nodes on the network by providing alternating periods for scheduling and transmission. A scheduling period begins at the termination of a successful transmission of a packet. Each node has a unique time delay into the scheduling period, when it may initiate transmission if the channel is idle. The scheduling period is terminated by the initiation of transmission and the next scheduling period begins at the end of that transmission.

In the Modified BRAM, the nodes of the network are placed into several priority levels. Starting with node 1 of priority level 1 (highest priority level), each node is given a slot to transmit. This node is followed by the other nodes of priority level 1 in an increasing order. Once level 1 nodes have completed, one node from the priority level 2 is allowed to transmit.

The Modified CSMA protocol is proposed to bound the delay for the lower priority frames. Bounded delay can be achieved by increasing the access time of higher priority nodes more than lower priority nodes, after transmitting a packet.

The generalized CSMA/CD protocol with dynamic priority combines the contention mode in the idle state of the channel and reservation mode in the busy state of the channel. In the idle state of the channel, the protocol operates in the contention mode, i.e., it reveals small access delays as the well known CSMA-CD protocol for low traffic. In the heavy traffic region, the protocol operates effectively in the reservation mode, with stations transmitting according to a deterministic access scheme. The access rights are implemented through staggered delay time after a successful transmission and are dynamically changed upon broadcast acknowledgement. The protocol offers several options for dynamic adjustment of access priorities, depending on system state or specific performance requirements.

The main difference between all the methods discussed with respect to CEBUS based on EIA standards which utilizes the CSMA/CD is that in CEBUS, each node in the network acts as an independent agent to regulate the channel access while for all other methods each node in the network has an index number, and all the nodes in the network are given a chance to transmit in an ascending order. Although all nodes have equal opportunity at channel access in CEBUS, some nodes may have to defer their channel access several times before transmitting, while other nodes may be able to access the medium several times in succession. To prevent excessive delay and to impose more stringent bounds on delay on certain consumer applications than others such as the security alarms, CEBUS standards specify, as mentioned previously, three levels of priority HIGH, STANDARD and DEFERRED where channel access delay is varied with respect to the priority of the packets.

In the area of performance evaluation, Pakkam and Manikopoulos have studied the CEBUS performance based on Power Line (PL) without acknowledgment by measuring delay vs. offered load and throughput for a number of high priority nodes[12]. Markwalter et. al. have investigated the CEBUS design using a prototype router implemented with computer hardware[13]. The priority assignment of the packets used, however, was limited only to HIGH in order to keep channel access delays consistent.

The main purpose of this thesis is to implement CEBUS using Power Line with ACKNOWLEDGMENT which has not been investigated before and is regarded as a realistic and more reliable CEBUS implementation. The performance of such a CEBUS will be evaluated through a series of simulation experiments involving quantities such as package delay, message delay, and throughput with varying parameters such as HIGH, STANDARD, and DEFERRED priorities, package size and message size, offered load.

As knowledge of the CEBUS architecture and protocol is essential for a good understanding of this project, the following chapter, Chapter 2 will be devoted to the description of the CEBUS architecture and protocol. Chapter 3 presents the simulation model of the CEBUS and the simulator itself. Results of simulation experiments using the simulator will be described and discussed in Chapter 4. Conclusions will be summarized in Chapter 5.

CHAPTER 2

ARCHITECTURE AND PROTOCOL OF THE CEBUS

2.1 CEBUS Architecture

The CEBUS, as briefly mentioned in Chapter 1, is a local area network which provides a standardized communication facility for the exchange of control information among devices and services within a home/office building. Primary consideration in the design of the CEBUS is low cost, ease of operation and retrofit, and versatility with both distributed and centralized control. Consideration is also given to the expandability over time as new media and new technologies are adopted. The CEBUS addresses these considerations by providing a standard communications interface to a number of different media (power line, twisted pair, fiber optic, coaxial cable, RF, and infrared). In particular, use of the existing 60Hz power line as a communications medium in consumer applications reduces the cost of installation of inter-room wiring between devices.

The Open Systems Interconnect (OSI) model is employed in the design of the CEBUS architecture for data communications/interchange. Four of the seven OSI layers[1,2] are used in the CEBUS as shown in Figure 2.1 wherein the Data Link Layer is divided into the Medium Access Control (MAC) Sublayer and the Logical Link Control (LLC) Sublayer. By enabling different Medium Access Control Sublayers to be interchanged with a universal Logical Link Control Sublayer, different channel access techniques are permitted. Thus, the operation of the Data Link Layer is made more flexible. Some of the functionality associated with the Transport Layer is built in to the CEBUS Network and Application Layers. Since the Session and the Presentation Layers of the OSI model are not required for the CEBUS, they are omitted to minimize both packet length and device complexity.

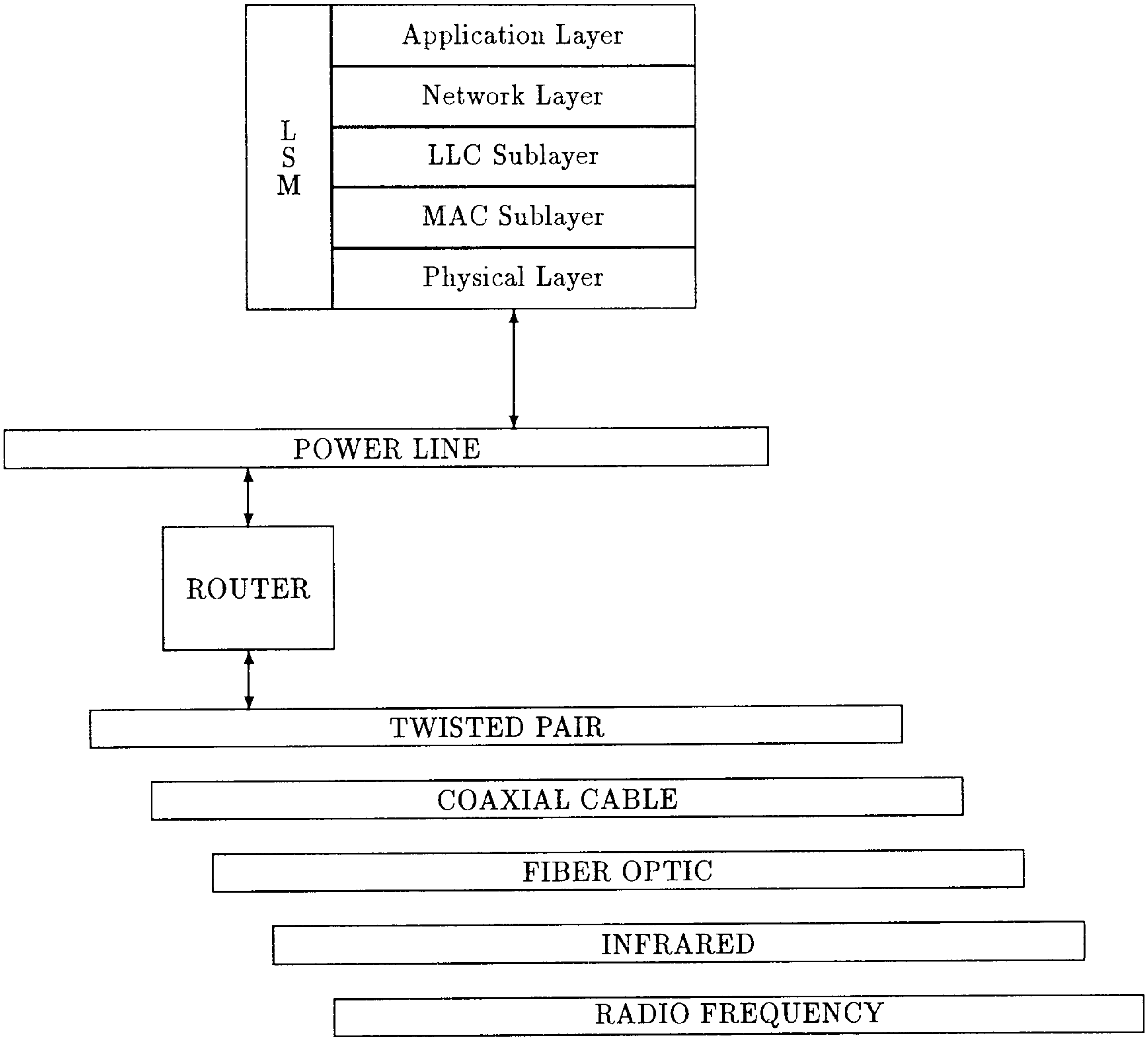

Figure 2.1 CEBUS architecture

2.1.1 Layer System Management

The Layer System Management (LSM) provides an interface mechanism between non-adjacent layers, initializes and maintains the peer-to-peer protocol of each of the layers/sublayers. Conceptually, it is adjacent to each of the layers/sublayers and performs various network administrative functions such as reading and setting parameter values in different sublayer and resetting Layer entity to a known state. It also notifies different layer/sublayer of significant events in the LSM or in the other layers/sublayers of the node.

2.1.2 Physical Layer

The Physical Layer provides the Direct physical connection to the communication medium for transmission and reception of data symbols is provided by the Physical Layer. Each node has a separate Physical Layer specification. The symbols of a frame are given serially to the Power Line Symbol Encoding Sublayer (Physical Layer) for transmission. The signal encoding for the Power Line will be Non Return to Zero(NRZ), and Pulse Width Encoding (PWE) using the symbols of 1, 0, EOF, and EOP, where EOF represents the End of Field symbol inserted between two fields in a frame and the EOP stands for the End of Packet which terminates the last field in a frame. The encoding of the symbols will be performed using the SUPERIOR and INFERIOR states on the PL medium. During the preamble portion of the CEBUS message, the presence of the frequency swept carrier on the PL will represent the SUPERIOR state, and the absence of the carrier will represent the INFERIOR state as shown in Figure 2.2(a). However, during the non-preamble portion of the message, the frequency swept carrier is continually transmitted and encodes the different symbols by reversing the phase of the carrier sweep(at the start of a new sweep) as shown in Figure 2.2(b). The encoding of the symbol is strictly

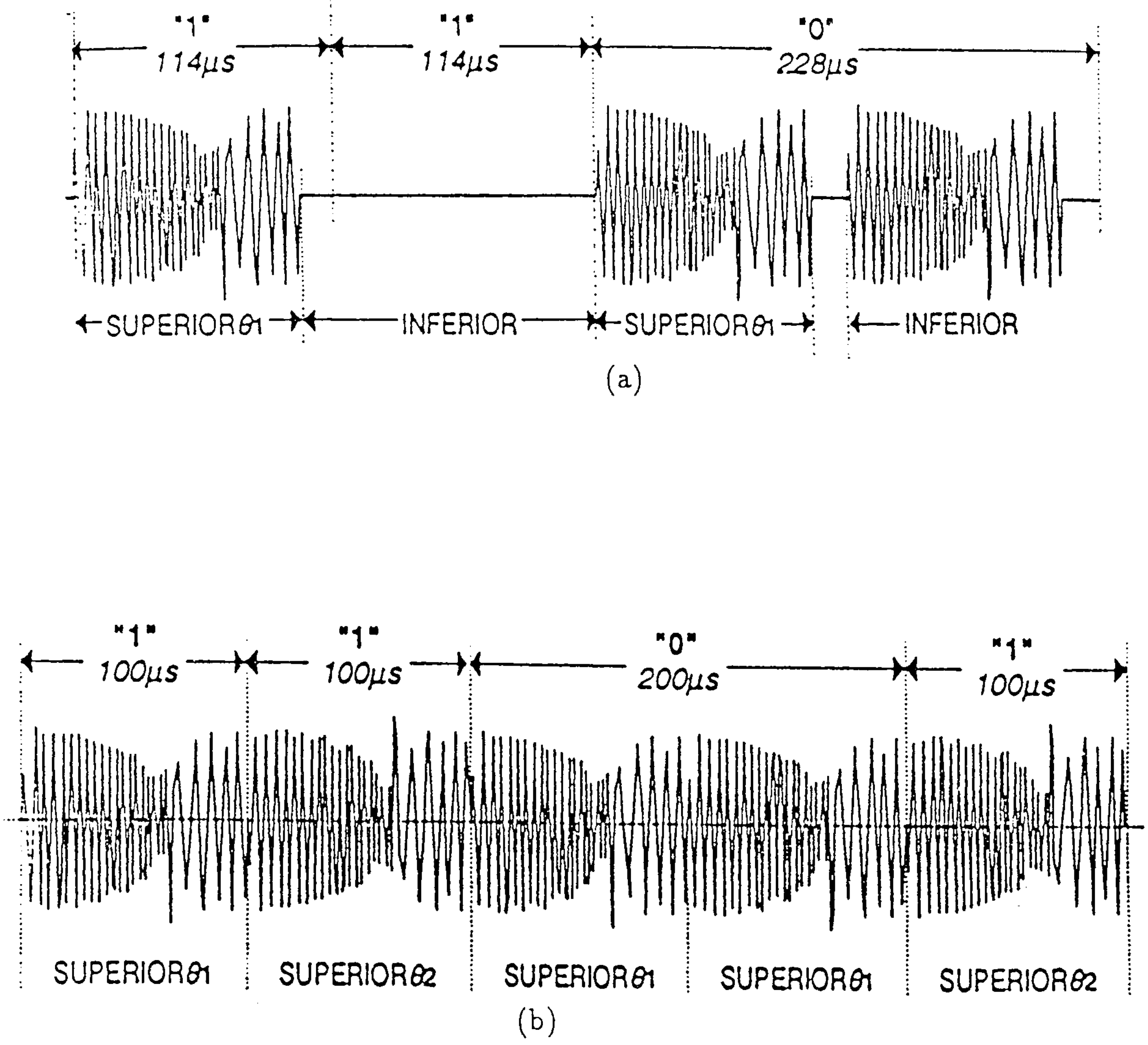

Figure 2.2: PL control channel preamble encoding example (a), and non-preamble encoding example (b).

Table 2.1 Data rate

Physical Medium	Data Rate
Power Line (PL)	10,000 ONE bits/sec
Twisted pair wire (TP)	10,000 ONE bits/sec
Coaxial cable (CX)	10,000 ONE bits/sec

related to the time the INFERIOR or SUPERIOR state remains on the media, not whether the INFERIOR or SUPERIOR state is used. For Power Line network,

The time needed to transmit the shortest symbol (ONE) will be defined as the "Unit Symbol Time" (UST). The shortest symbol is ONE which is 100 μs. Symbol ZERO of 200 μs then has 2 UST. Symbol time for EOF and EOP are 3 and 4 UST, respectively. The data rate for Power Line is 10,000 ONE bits per second $\pm$ 0.1% over the operating temperature and humidity range of the PL devices. To make detection of the preamble easier, the UST is longer during the preamble than during the message body. During the preamble, ONE, ZERO and EOF are 114 μs, 228 μs and 800 μs $\pm$ 0.1%, respectively. Table 2.1 shows the data rate for 3 different media. The rate for optical fiber can be more than 50 Mb/s. If lasers and single-mode fibers are used, the range of bandwidth can be in the range of Gb/s.

2.1.3 Medium Access Control Sublayer

The Medium Access Control (MAC) Sublayer interacts with the Physical Layer to monitor the channel and control data reception and transmission. Through the use of the frame check sequence the validation of received frames is also performed. Upon receiving a frame, the MAC Sublayer disassembles the frame and passes the Logical Link Control Sublayer Protocol Data Unit (LPDU) up to the LLC Sublayer. The MAC Sublayer handles the majority of the Data Link Layer functionality. Communication between user processes in the CEBUS model involves the following sequence of data transfer. A message originating from a user process is passed to the Application layer for incorporation into an Application Layer header, i.e. an Application Protocol Data Unit (APDU). The APDU is then passed down to the Network layer. A Network layer header, i.e., a Network Protocol Data Unit (NPDU) is generated from the control information and tagged onto the APDU. The message is then passed down through the Data link layer (MAC and LLC), to Physical layer for transmission. In passing data between adjacent layers, no layer is allowed to alter the Protocol Data Unit (PDU) passed to it from the layer above. A PDU must be handled as a unit entity.

The communication of the MAC with the LLC Sublayer is by means of an interlayer interface. Through the interface, the MAC Sublayer performs the functions of transmitting and receiving the LPDU. Only one type of service is offered, unacknowledged connectionless service. To transmit an LPDU, the MAC Sublayer first incorporates the LPDU into a MAC Sublayer Protocol Data Unit (MPDU) and then follows the Carrier Sense Multiple Access/with Collision Detection and Collision Resolution (CSMA/CDCR) channel access protocol prior to transmission through the Physical Layer.

2.1.4 Logical Link Control Sublayer

The Logical Link Control (LLC) Sublayer offers two types of services for the the transmission and reception of NPDU: acknowledged and unacknowledged connectionless service. Acknowledged service makes use of the IACK mechanism. When a frame is received by the MAC Sublayer, the LPDU is removed and passed to the LLC Sublayer. The LLC header is then removed from the LPDU and the remaining NPDU is passed up to the Network Layer. If acknowledged service is used, the LLC Sublayer commands the MAC Sublayer to generate an IACK frame and send it out onto the network.

To transmit a packet, an LPDU is generated and passed down to the MAC Sublayer along with its associated control parameters. The MAC Sublayer takes the responsibility of assembling and getting the frame onto the channel. In case of acknowledged service, the LLC Sublayer also takes the responsibility of receiving the IACK frame.

2.1.5 Network Layer

The Network Layer performs routing of the NPDUs between different media through specialized devices known as routers to be described briefly later. Except for the segmentation and network connections, Network Layer is responsible for all the functions described in the OSI reference model. The design of the CEBUS places the segmentation function in the Application Layer with the flow control of the segments handled by the Network Layer. Much of the complexity is forced into the routers, which are few in number's compared to the nodes. Connectionless service is employed at the Network Layer of the CEBUS.

2.1.6 Application Layer

A new language called Common Application Language (CAL) is provided by the CEBUS Application Layer through which product manufacturers may construct device-specific control functions. Communication of consumer products therefore have a standard interface into the network. The Application Layer supports a command and response protocol that can be used to guarantee end-to-end message delivery and also performs segmentation of a large message into smaller packages. A APDU header similar to those found in the lower layer is added to the front of the CAL commands before being passed along .

2.1.7 Routers for CEBUS

The connection between the aforementioned six types of different media in the CEBUS network is accomplished by the use of the CEBUS routers. A brief description of a router is provided next although the project described in this thesis does not involve a router.

A router can receive packets from different media, buffer the packets, and decide whether or not to forward each packet onto the next medium, depending on the contents of the header fields within each of the packet. Routers should also communicate with each other to maintain the network topology in a tree structure. To develop a tree structure, the network topology is subdivided into logical topology (LT) and physical topology (PT).

LT decides how the network media are logically interconnected. Before a new router may handle network traffic, it transmits a HELLO packet identifying itself to its neighboring router. Existing routers then respond with their own HELLO packets. Any topological loops detected through this process are eliminated by logically disabling router connections. HELLO packets are periodically circulated to maintain intermedia connections.

PT describes the allowed physical interconnections between the network media. Since Power Line will normally be the main medium in CEBUS networks, it is viewed as the trunk of the tree structure and other media connected to the Power Line are treated as the branches of the trunk. Each branch is connected through a router to the Power Line trunk.

The CEBUS router architecture is designed in the same way as nodes. It, however, has two LLC sublayers, two MAC Sublayers, and two Physical Layers. Difference between node and router MAC Sublayer services are a consequence of the router's task of properly forwarding packets through the network, rather than simply transmitting and receiving packets. The peer-to-peer protocols between the corre-

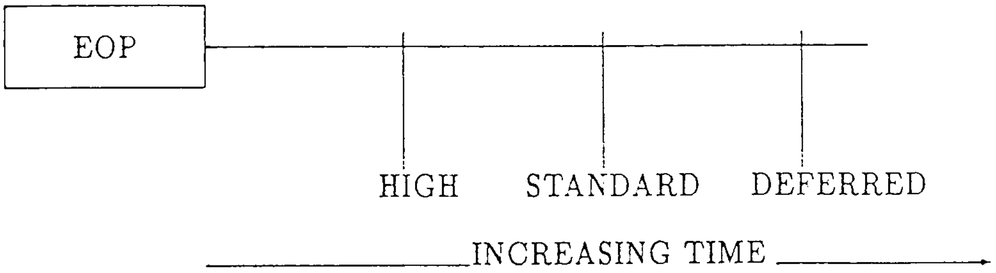

Figure 2.3 Priority delay

sponding router layers are identical to the node protocols. Router LSM initializes and maintains the peer-to-peer protocol of each layer and provides an interface between non-adjacent layers, and manages issues which relates to the system or the network as a whole, such as maintaining a correct network topology.

2.2 The CEBUS Protocol

2.2.1 Prioritization

To eliminate the interference of lower priority messages to higher priority messages, each message is assigned a priority level which is passed down from the Network Layer and denotes the relative level of importance of the message. The effect of the priority level is to delay the transmission of a message for an additional period of time and the amount of delay differs, depending on the level of priority. With a shorter delay, higher priority messages have a greater chance of obtaining control of the channel.

In CEBUS protocol, there are three priority levels named HIGH, STANDARD, and DEFERRED. When HIGH priority has been assigned to a message, it will not allow any additional delay to the transmission of that message. A STANDARD priority will impose a 4 unit symbol times (USTs) of additional delay to a message transmission, while a DEFERRED message will be delayed for an additional 8 USTs. These delays are in addition to the unit symbol times of mandatory channel quiet (minimum wait time of 10 UST). Figure 2.3 illustrates these priority delays following the end of a frame (EOP symbol). This scheme allows nodes with higher priority frames to seize the channel before nodes with lower priority frames.

2.3 Round-robin Queueing and Scheduling

It is apparent that contention may still occur between nodes at the same priority level although deference and prioritization have been employed to reduce the probability for conflict over the use of the channel. To ensure contending nodes each have an equal opportunity for channel access, a "round-robin queueing" method within each priority level is used. Queueing allows more orderly access to the medium among nodes transmitting frames of the same priority.

QUEUED STATE: Once a node successfully accesses the medium and transmits a frame, it becomes QUEUED. For QUEUED nodes, an extra delay is required for medium access relative to UNQUEUED nodes having frames of the same priority. This additional delay ensures that UNQUEUED nodes waiting with frames of given priority will access the medium before any QUEUED nodes with frames of the same priority. Figure 2.4 illustrates the queueing process in the CEBUS. The bold lines show the randomization intervals of 0 to 4 UST.

UNQUEUED STATE: Unqueued state occurs in one of the following two cases:

- If the node or station has no message to send or the medium is sensed idle for the maximum channel access time of 26 UST slots; or

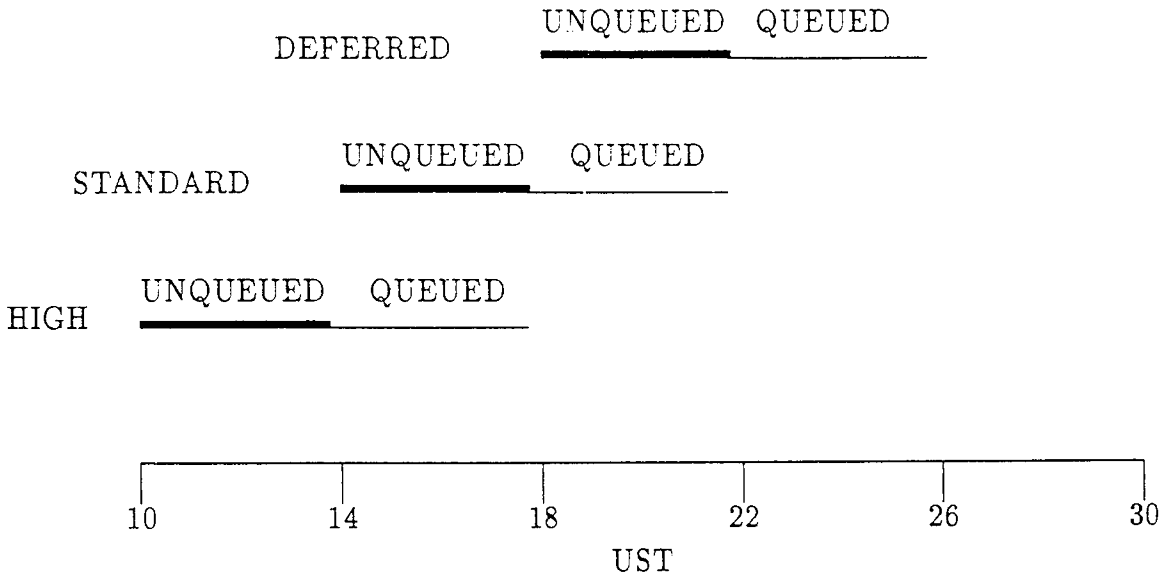

Figure 2.4 Priority queueing

- If none of the QUEUED nodes complete a transmission during the following 4 UST slots.

An UNQUEUED node waiting with an HIGH priority frame starts contention any where between 0 to 4 UST immediately after the end of packet symbol. If an UNQUEUED node loses the contention the node remains UNQUEUED. QUEUED High priority nodes must wait 4 UST plus 0-4 UST to start contention. They become UNQUEUED when losing the contention. An extra 4 UST are assigned to the corresponding STANDARD priority. Comparing with the STANDARD priority, the DEFERRED priority has yet another extra 4 UST.

The randomization interval of 0 to 4 UST is employed to further reduce the probability of contention because more than one node may be in the same priority level and queueing state. This randomization sets the transmission start-time for each contention node into four distinct time slots (namely, 0, 1, 2, 3 UST slots) as shown in Figure 2.6. This reduces the possibility that two or more nodes, which have fallen into the same priority/queueing time slot and have had to defer or abort

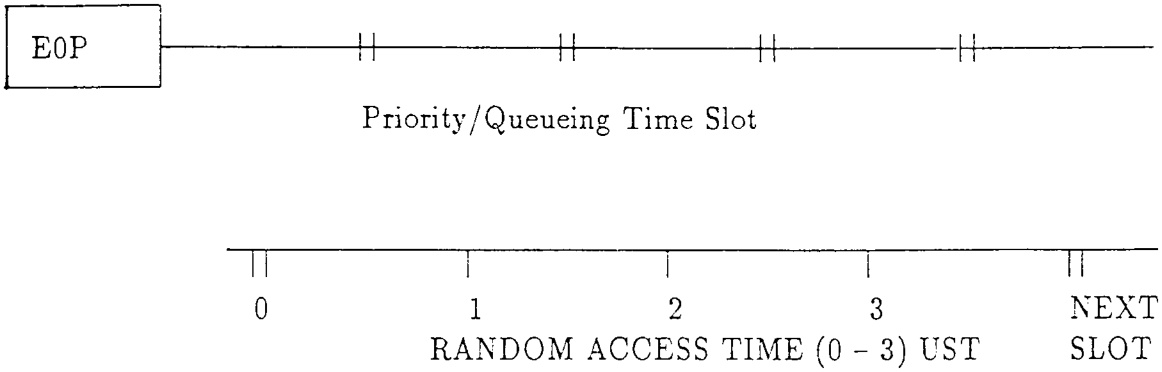

Figure 2.5 Random access time

their transmissions, try to access the channel in the same time slot during the next opportunity for transmission. Transmission starts at one of four random access time slots, within a particular priority/queueing time slot.

2.4 Contention Detection and Resolution

A non-zero probability of contention may still exists when two or more nodes try to transmit simultaneously in the same time slot although the Data Link layers of all nodes follow the channel access method aforementioned, which is designed to avoid conflict over the use of the medium. Typically, the conflict between nodes is resolved during the preamble. The value of the Preamble Field is a pseudo-random 8-bit sequence which activates contention detection and resolution as part of the channel access protocol. For instance, after two nodes have a collision, they will send a random bit sequence of SUPERIOR and/or INFERIOR states by the Physical Layers to the medium. The contention is solved during the Preamble slot without losing any information when one of the colliding nodes transmits a SUPERIOR state and the other transmits a INFERIOR state. However, when two or more nodes transmit simultaneously during some part of the frame past the preamble, interference occurs in one or more of the frames, and data is lost. Such interference results in an incompletely received frame. Contention past the preamble may also occur as a result of a node not following the channel access protocol. In this case, the faulty node breaks into the middle of the transmission of another node, causing its transmission to be aborted.

2.5 Immediate Acknowledgment

The immediate Acknowledgment (IACK) provides the mechanism for the transmitting node to determine if its message is transmitted successfully through the medium. Obviously, this feature is important as it results in a more reliable network performance. But, it is at the expense of the longer delays. Past performance evaluation reported in the literature did not include IACK. In this thesis, both acknowledged and unacknowledged PL CEBUS network performances have been successfully implemented.

IACK is activated when acknowledged connectionless service is requested by the Network Layer. The receiving node forms an IACK frame when the message frame is properly received and an acknowledgment is requested. The IACK frame is delivered to the local medium within 2 UST after the end of EOP symbol of the originating frame. All other nodes are in the minimum wait state for 10 UST. By immediately taking control of the channel (transmitting while the originating node still "owns" the channel), the receiving node is assured of sending the IACK without having to contend for the channel. Contention during IACK transmission constitutes a failure of the Data Link Layer protocol and will cause the receiving node to abort the IACK. Also, noise received during the time between the end of the originating frame and the transmission of IACK will prevent the IACK from ever beginning.

In acknowledged case, the transmitting (originating) node expects to hear the beginning of the IACK preamble within 6 unit symbol times of the end of the EOP symbol of its frame. The incoming fields are parsed to ensure that a fragment is not received when the originating node receives the IACK frame. The Frame Check Sequence field (FCS) in IACK frame format contains an 8 bit checksum value. The checksum value is obtained by summing each 8 bit field from the frame (excluding the preamble) with carries discarded. The two's complement operation is performed on the checksum and this final value is passed to the FCS field. If the received frame format is correct, the checksum operation is performed to verify the data. A resulting sum of zero indicates a valid received IACK frame. When the IACK is correctly received, its preamble and FCS fields are discarded and the control field is processed within the Data Link Layer. Three features clearly distinguish an IACK from an originating frame: its time of arrival, format of the frame (the number of EOF fields), and the control field (Packet Type). In CEBUS, the receiving node's Data Link Layer transmits the IACK during the minimum wait time for all other nodes. Thus, channel access is different for an IACK than for an originating frame. Also, the originating node's Data Link Layer is aware of an incoming IACK frame by virtue of its arrival time. The advantage is that the IACK may be transmitted free of contention.

CHAPTER 3

SIMULATION MODEL

3.1 The Simulator

The simulator is briefly described and the definitions are introduced which provide the foundation for the analysis and discussion of the simulation results in the following sections.

The simulator employed for the system and protocol modeling in the following experiments was written in C language using the C-Library functions provided by LANSF [14]. It can be modified to simulate the CEBUS architecture proposed in the EIA standard released in September, 1992 [2]. It is a configurable simulator designed to model communication networks. The attributes of a communication network that can be specified in LANSF are divided into two categories. The first category contains static elements, i.e. the system architecture and topology. The second category consists of dynamic attributes that describe the temporal behavior of the modeled system, i.e. the traffic patterns, and performance measures. The simulation involves two tasks, system and protocol modeling and network configuration. The CEBUS system and protocol modeling requires C programming using the C-Library functions provided by LANSF while the network configuration does not involve C programming. It is specified in a data file which is interpreted by the system and protocol designed. There are four program files needed to interface LANSF and the CEBUS network, including (i) protocol.c, (ii) protocol.h, (iii) options.h, and (iv) input data file.

The *protocol.c* specifies the executable part of the protocol specification and functions which represent protocol processes executed by stations (nodes). It also contains other two subroutines that must be included with the protocol module. The first, the *in_protocol*, initializes the simulator and read the values of the global protocol-specific parameters. The protocol-specific parameters are read in the same order in which they occur in the input data file using three functions, the *read_integer*, *read_real*, and *read_big*. The second extension function that must be defined in protocol.c is the output file *out_protocol*. This output file contains the output results and the the protocol-specific input parameters. The program file protocol.c consists of a number of simulated processes running at each node specified in the input file. The execution of these C-functions is scheduled by the event handlers. Processes are scheduled by either the built-in LANSF servers such as TIMER and BUS events or signals from other processes. The signalling mechanism provides a method for inter-process communication, and can be extended to simulate layered protocols as processes.

The definitions of protocol-specific symbolic constants and the declarations of non-standard station attributes are contained in the *protocol.h* file. The contents of protocol.h are inserted into the declaration of the structure STATION [14]. All variables defined in *protocol.h* are actually declared as attributes of STATION and made visible to *protocol.c*. A copy of this file must be presented in each protocol directory.

The *options.h* files contains the local options such as precision of numbers, the type of port variables representing port transmission rates, the length of additional information carried by messages and packets, the type of transmission link, and the number of moments to be calculated for standard statistics. A copy of this file should be present in each protocol subdirectory.

The input data file starts with time section and configuration section which define the network backbone. It starts with number of stations followed by specification of the number of ports for each station, link number and type, total number of ports and its transmission rate, distance matrix describing the distance between the nodes, the number of messages, message length, mean interarrival time, the number

Table **3.1** Service primitives

Transmit	Receive
N_DATA. request	N_DATA. indication
L_DATA. request	L_DATA. indication
M_DATA. request	M_DATA. indication
L_ACK_DATA. request	L_ACK_DATA. indication

of senders, receivers and optional flood group or broadcast type messages. The next section of the input data file consists of protocol-specific parameters. To read it, LANSF calls the function *in_protocol* from program file *protocol.c* followed by exit conditions, namely the total number of messages to be generated, simulation time and CPU time limit.

As an example of illustration, the following describes the transmitter of a station (node).

The transmitter function can be best explained by the service primitives shown in Table 3.1 as described in [2] which provides the interlayer communications. We consider a message fetched into the station (node) buffer with its length defined in the input data file, the Application Layer sends a signal (N_DATA.request) to the Network Layer which in turn adds NPDU to the packet and passes it to the layer below. While passing the packet to the different layers, only pointers to data are passed through the layer rather than copying the data several times. The channel access function of the MAC sublayer is explained by examining the codes shown below.

```
case CHANNEL_ACCESS:
    priority = PACKET→type;
    Access_delay = priority_delay[priority] + random_interval();
    last_silence = last_eoa_sensed(BUS);
    if(def(last_silence)) {
```

```
idle_period = minus(current_time, last_silence);
if(geq(idle_period, Access_delay))
continue_at (TRANSMIT_PREAMBLE); }
```

The type of the message 0, 1, 2 for HIGH, STANDARD, and DEFERRED is obtained by accessing the packet_buffer "PACKET," which stores the packet to be transmitted. The Access_delay of the respective priority message is varied by accessing the array priority_ delay [0, 4, 8]. For example, if the priority level of the message is 1 (Standard priority), the channel access delay is obtained from the array priority_delay[2], which is 4 unit symbol time. Function random_ interval returns a random delay time of 0 to 3 UST to avoid contention among the nodes of same priority. The subroutine last_eoa_sensed(BUS) returns the time, when the last activity was heard in the channel. Subroutine def(last_silence) checks if the channel is idle, then calculates the idle_period by subtracting the current_time (time measured since the beginning of the protocol execution), from the time last activity sensed in the channel. If the idle_period is greater than the Access_delay, the packet is transmitted at once, else the channel access is delayed till idle_period is equal to Access_delay time.

After successful transmission, all the nodes wait for an additional delay of 10 UST, before accessing the channel. After a packet is transmitted, internal_signal (M_DATA. confirm) is generated and sent to the LLC sublayer, just to make sure that the packet is transmitted. This does not assure the proper reception of packet at the destination node. The function mac_receiver() passes the packet to LLC sublayer by sending internal_signal (M_DATA. indication), which in turn removes the header information and passes the packet to the layer above by sending internal_signal (L_DATA. indication).

In the case of acknowledgment, confirming does not take place before the acknowledgment is complete. An IACK message is sent from the receiving to the originating station within the 2 UST of the end of the EOP symbol of the originating frame. During this time the station still owns the channel so there is no contention during the IACK transmission.

3.2 Performance Measures and Definitions

The most important measure of the network performance is delay of signal transmission. This involves two types of delays. The first one is the packet delay and the second is the message delay. Another important measure is the channel throughput. The number of message of successful transmission is also an important information regarding the network performance.

1. Packet Delay: it is defined as the time elapsed from the moment a packet becomes ready for transmission in the originating station (node) to the moment the packet is successfully received by a station (node). It does not involve the queueing time.

2. Message Delay: it is measured as the time elapsed from the moment a message is queued at the originating station (node) to the moment the entire message(all its packets) is successfully received by a station (node). Message delay includes the message queueing time.

3. Channel Throughput or Throughput: it is calculated as the ratio of the total number of information bits successfully transmitted through the channel to the simulation time. Note the term "information bit" means a bit belonging to the information part of a packet, i.e. a message bit. Packet headers and trailers do not count.

4. Message Throughput: it is measured as the ratio of the total number of bits received at the destination address to the number of bits generated at the source.

When a message is chopped into packets of different lengths, the following calculation shows how the message delay $d_m(M)$ and $d_p(P)$ can be simulated. Considering a sequence of messages $M^1, \ldots, M^n$ and assuming message M^j consists of packets $P_1^j, \ldots, P_{k_j}^j$ with lengths $l_1^j, \ldots, l_{k_j}^j$ respectively. Let $l^j = \Sigma_{i=1}^{k_j} l_i^j$ denote the length of M^j. Message M^j was queued at the sender at time tq^j, its i'th packet P_i^j became ready for transmission at tt_i^j and was completely received by the target station at tr_i^j. The message delay for M^j and packet delay for P_i^j are:

$$d_m(M^j) = tr_{k_j}^k - tq^j$$

$$d_p(P_i^j) = tr_i^j - tt_i^j$$

The time when a packet becomes ready for transmission tt_i^j is determined as the maximum of the following two values.

1. the time when the buffer, the packet acquired into, was last released.
2. the time when the message, the packet acquired from, was queued at the station.

The distribution parameters of the random variable representing the message delay of multiple messages transmitted over the network are calculated assuming that the random variable consists of discrete samples, namely, the message delays of particular messages. For instance, the average message delay for the n messages $M^1, \ldots, M^n$ is computed as:

$$d_m^a(M^1, \ldots, M^n) = \frac{\Sigma_{i=1}^n d_m(M^i)}{n}$$

The absolute packet delay is interpreted in a similar way and the formula for determining the average delay is:

$$d_p^a(< P_i^j, \ldots, P_{k_j}^j >_{j=1}^n) = \frac{\Sigma_{j=1}^n \Sigma_{i=1}^{k_j} d_p(P_i^j)}{\Sigma_{j=1}^n k_j}$$

Table 3.2 Simulation parameters

Total No. of Message	5,000
Types of Priority	3 (HIGH, STANDARD and DEFERRED)
Total No. of Station	30
No. of HIGH Station	10
No. of STANDARD Station	10
No. of DEFERRED Station	10
Data Rate	10 Kb/s
1 UST	100 μs ± 100 ns
Duration of Symbol ONE	100 μs ± 100 ns
Duration of Symbol ZERO	200 μs ± 200 ns
Duration of Symbol EOF	300 μs ± 300 ns
Duration of Symbol EOP	400 μs ± 400 ns

For simplicity, message delay, instead of mean message delay will be used to describe the statistical simulation results. Similar simplicity is used for packet delay and others.

In summary, the model for the simulation of PL CEBUS network performance can be characterized by the parameters shown in Table 3.2.

CHAPTER 4

ANALYSIS AND DISCUSSION OF SIMULATION RESULTS

4.1 General Description of the Simulation Experiments

The performance of the CEBUS implemented with power line has been simulated concerning the message delay, packet delay, channel throughput, and message throughput under different normalized offered load, G, over a large range.

Three different cases have been studied involving three different packet bit sizes, i.e., 600, 300, and 100 bits where the 600 bits packet is around the allowed maximum packet size and the 100 bits packet is near the minimum packet size as specified in the CEBUS standards.

Because the packet size, especially the 100 bits packet, is not too much larger than the IACK frame which is 24 bits, there is no need to chop it into smaller sizes to observe the performance difference induced by the use of IACK as will be seen later. Indeed, even when the packet size is 300 bits noticeable difference already appears in the performance with and without IACK.

The following studies, therefore, use equal message and packet length to reveal the queueing time effect which was the approach used in other literature reports [15, 16]. A set of data will be shown to reveal all the performance parameters' dependence on the normalized offered load G, which is defined as the total offered load normalized by the channel capacity C. The total offered load is $(\lambda_H \mathrm{T}_H + \lambda_S \mathrm{T}_S + \lambda_D \mathrm{T}_D)$ where λ_i's and T_i's(i=H, S, D for HIGH, STANDARD, DEFERRED messages, respectively) stand for the arriving rate of packet and packet length for the three types of messages, respectively. The λ_i's are assumed to follow the symmetrical Poisson distribution and so λ_i's are equal to each other

$$\lambda_H = \lambda_S = \lambda_D = \lambda \tag{4.1}$$

The normalized load is therefore

$$G = \frac{3\lambda T}{C} \tag{4.2}$$

Comparative simulation experiments have also been done to compare the performance parameters such as message delay, packet delay, message throughput, and channel throughput between unacknowledged and acknowledged cases.

As one of the main performance concerns of the CEBUS is the delay incurred by Standard and Deferred priority packets in the presence of High priority traffic, the delays will be shown first. Two types of delays have been measured as part of the simulation, the packet and message delays. The packet delay is measured from the time a node has a packet available for transmission, to the time when that packet has reached the destination node and has been accepted. It does not include queueing time at the node. The message delay is measured from the time a message is queued at the node to the time it is delivered to the destination. A series of offered load versus delay measurements were taken for the CEBUS. These results show typical delays of the CEBUS, the region in which overloading starts, and the effect of High priority traffic on the CEBUS performance.

The simulation was run for various traffic patterns while the throughput was varied by increasing the offered load. The exponentially distributed traffic consists of 1/3 of the traffic from each of the three priority nodes (HIGH, STANDARD, DEFERRED). All the simulations were run for a total of 5,000 messages. The total number of stations (nodes) are 30 with 10 for each priority.

Packet lengths of 600, 300, and 100 bits have been considered for the following simulation experiments. These three packet lengths are chosen because, as previously discussed, the 600 and 100 bits are already around the maximum and the minimum for PL CEBUS, respectively. Message length is set equal to the packet length for the reasons mentioned in the previous section. A set of comparisons are shown as follows. Notice that only one figure for each comparison is shown in this chapter with the rest of the figures shown in Appendix A in order to meet the thesis format.

4.2 CEBUS Performance With and Without IACK: Case of 600 Bits

The performance characteristics of the CEBUS *with* and *without* acknowledgment are shown in Figure 4.1, Figure A., and Figure A.2 for message delay, packet delay, and message throughput, respectively.

For the 600-bit message shown in Figure 4.1, the difference between the acknowledged and unacknowledged message delays is very small. This can be explained by the fact that the IACK frame is transmitted contention-free and by the fact that the IACK frame length is only 24 bits, much less than the 600 bits used in this study. Notice that the figure is plotted using log-log scale which has been used in other studies [15,16].

For the 600-bit packet shown in Figure A.1, the difference between the acknowledged and unacknowledged packet delays is also very small and is due to the same reason just mentioned.

For the message throughput vs. the offered load shown in Figure A.2, the difference between the acknowledged and unacknowledged packet delays is again very small. Notice that although the difference is very small, the acknowledged message throughputs for all of the three priorities are consistently seen to be on the lower side, in agreement with the notion that IACK frame causes the channel to be less effective.

4.3 CEBUS Performance With and Without IACK: Case of 300 Bits

4.3.1 Message Delay vs. Load

For the 300-bit message shown in Figure 4.2, the difference between the acknowledged and unacknowledged message delays for all of the three priorities are seen to increase

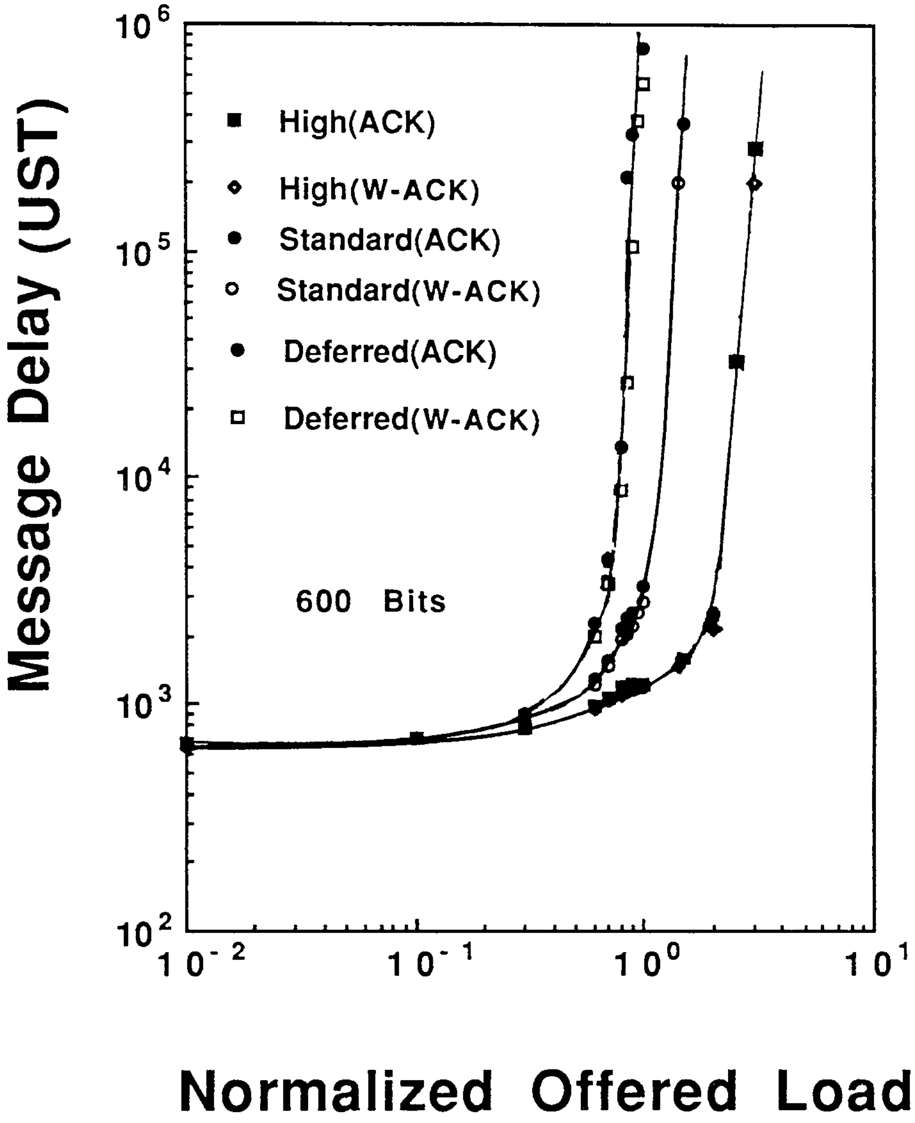

Figure 4.1: Message delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 600-bit message.

as compared to the case of 600 bits although the difference is not as easy to see as in a linear plot. For this reason, the message delay is replotted in linear scale as shown in the following sets of figures. It will be seen that linear scale shows the detail feature more clearly than the log-log plot.

The message delay vs. the offered load with and without acknowledgment is presented in Figure 4.3 for the HIGH priority. It is seen that there is a sharp increase when the offered load reaches 2. The sharp increase is due to the heavy load which when plotted in the log-log scale corresponds to the on-set point of the sharp increase shown in Figure 4.2.

The message delay vs. the offered load with and without acknowledgment is presented in Figure A.3 for the STANDARD priority. It is seen that there is a sharp increase when the offered load reaches 1. The STANDARD message is of lower priority when compared to the HIGH message which explains the earlier appearance of the sharp increase.

The message delay vs. the offered load with and without acknowledgment is presented in Figure A.4 for the DEFERRED priority. It is seen that there is a sharp increase when the offered load reaches 0.7. The DEFERRED message is of lower priority when compared to the STANDARD message which explains the ealier appearance of the sharp increase.

4.3.2 Packet Delay vs. Load

The packet delay vs. the offered load with and without acknowledgment is presented in Figure A.5 for the HIGH priority. The observation is consistent with the earlier ones, namely, the difference increases as the offered load is increased which is due to the increased delay between the time when a packet is ready for transmission to the time when the packet successfully gets into the channel and occupies the channel when IACK is involved in the channel transmission. However, a new

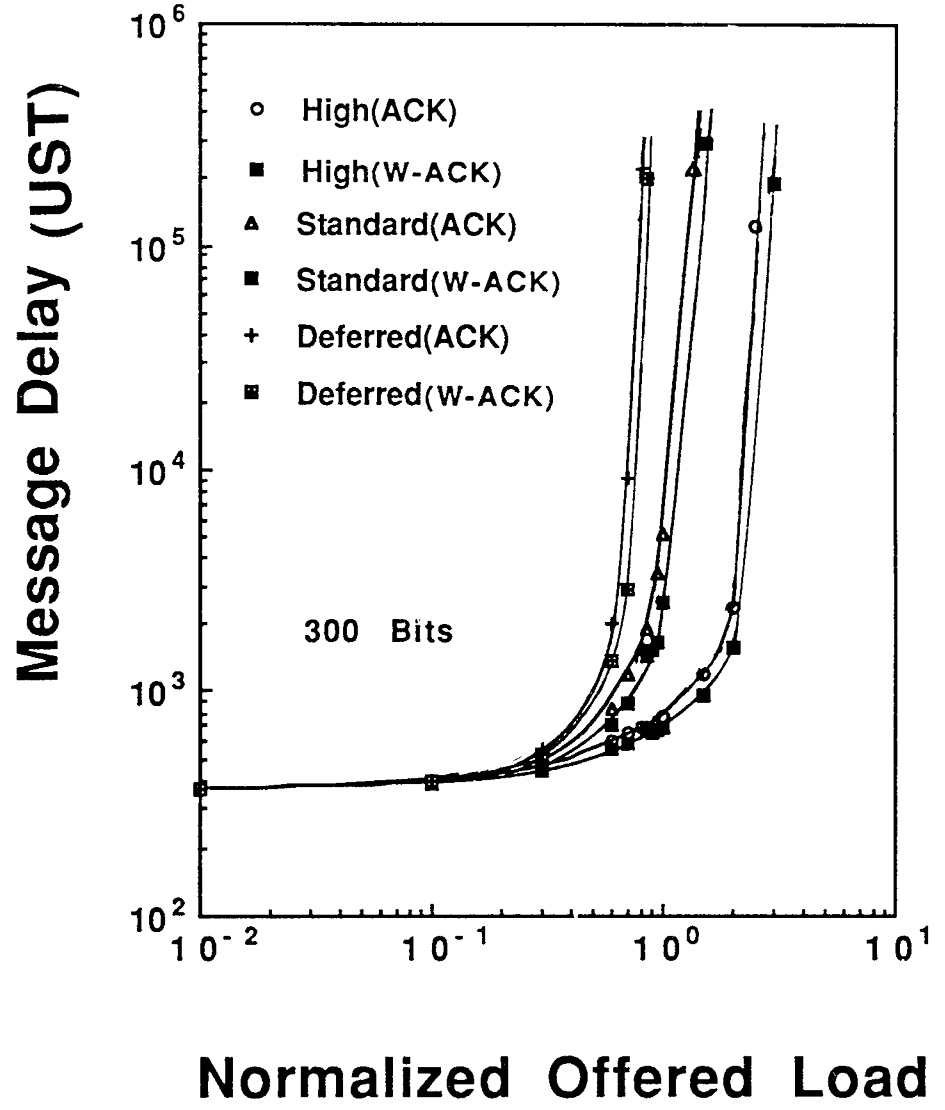

Figure 4.2: Message delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 300-bit message plotted in log-log scale.

feature, saturation of packet delay, appears when the load reaches 2.8 and 2.4 for unacknowledged and acknowledged cases, respectively. The tendency of saturation in nonacknowledged study has been reported before [15, 16]. The saturation is due to the fact that the message throughput at heavy loads already reaches zero for the STANDARD and DEFERRED messages, leaving only HIGH message in the channel. When the load reaches a limit, further HIGH packets generated at the source stations simply do not have chance to be transmitted and therefore do not count in the packet delay calculations. In other words, one can still try to offer billions of packets at the source side after the saturation point, they simply do not matter because there are already enough packets offered and are waiting before they that have little chance to content for the channel. When 5000 packets are transmitted through the channel the simulation is terminated because the statistical average is over 5000 packets for this simulation. The final saturation delays are different which again can be explained as a result of the extra occupation of the channel which reduces the maximum speed the channel can handle the transmission of information packets.

The situation is completely different for lower priority packets. Figure A.6 shows the acknowledged and unacknowledged packet delays for the STANDARD priority packets. The packet delay is much larger than that of HIGH priority case and it continues to increase after the offered load is more than 2. This continued increase has been reported even beyond the offered load equal to 2 [15, 16]. Here, the delay is not plotted beyond the load equal to 2 because it is noticed that the number of STANDARD priority message successfully transmitted through the channel already approaches zero when the offered load is 2. The statistical meaning simply does not exist for the packet delay when there is only one or two packets get through the channel.

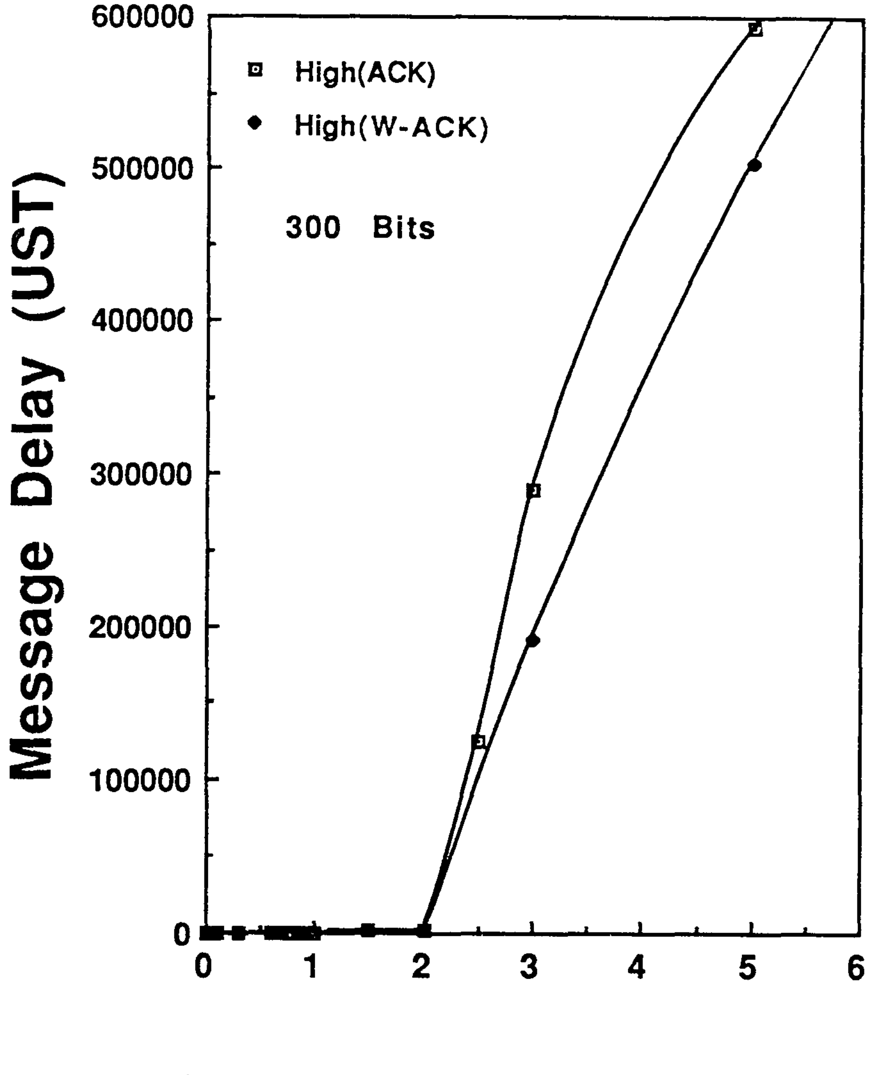

Figure 4.3: Message delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 300-bit HIGH priority message in linear scale.

For the DEFERRED priority shown in Figure A.7, it is seen that there is a sharp increase when the offered load reaches 0.7. The DEFERRED message is of lower priority when compared to the STANDARD message which explains the earlier appearance of the sharp increase.

4.3.3 Message Throughput vs. Load

The message throughput for the HIGH priority is shown in Figure 4.4 where it is clearly seen that the throughput starts to decrease when the offered load is equal to 2 which confirms the explanation for the message delay sharp increase when the offered load is equal to 2. The difference in the acknowledged and unacknowledged cases is due to the extra occupation of the channel by the IACK frame which does not count as an information message.

The message throughput for the STANDARD priority is shown in Figure A.8 where it is clearly seen that the throughput starts to decrease when the offered load is equal to 1 which again confirms the explanation for the message delay sharp increase when the offered load is equal to 1. The difference in the acknowledged and unacknowledged cases is due to the extra occupation of the channel by the IACK frame which does not count as an information message.

The message throughput for the DEFERRED priority is shown in Figure A.9 where it is clearly seen that the throughput starts to decrease when the offered load is equal to 0.7 which also confirms the explanation for the message delay sharp increase when the offered load is equal to 0.7.

4.4 CEBUS Performance With and Without IACK: Case of 100 Bits

4.4.1 Message Delay vs. Load

The message delay vs. the offered load with and without acknowledgment is presented in Figure 4.5 for the HIGH priority. It is seen that there is a sharp

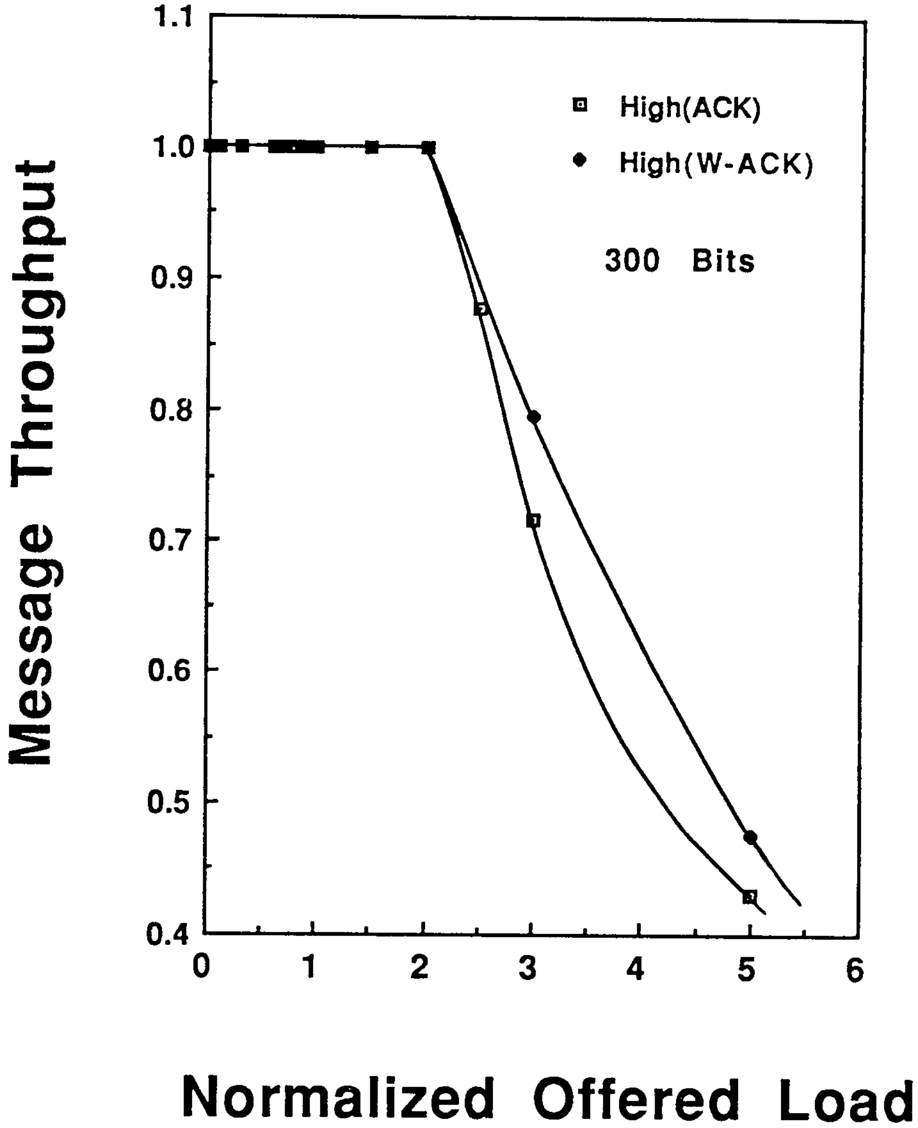

Figure 4.4: Message throughput vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 300-bit HIGH priority message.

increase when the offered load reaches around 1.5 which is smaller than the value of 2 for the case of 300 bits discussed earlier. The IACK induced delay is seen to be quite large when the offered load rises above 1.5 which is a result of the further increased sensing and contention time during the transmission of IACK frame.

The message delay vs. the offered load with and without acknowledgment is presented in Figure A.10 for the STANDARD priority. It is seen that there is a sharp increase when the offered load reaches around 0.8. The STANDARD message is of lower priority when compared to the HIGH message which explains the earlier appearance of the sharp increase. The IACK induced delay is seen to be even larger when the offered load rises above 0.8 which is a result of the substantially increased sensing and contention time during the transmission of IACK frame.

The message delay vs. the offered load with and without acknowledgment is presented in Figure A.11 for the DEFERRED priority. It is seen that there is a sharp increase when the offered load reaches around 0.3 to 0.4. The DEFERRED message is of the lowest priority which explains the earlier appearance of the sharp increase. The message delay difference for the DEFERRED priority when the load is equal to 0.6 is seen to be as large as the HIGH message delay difference for the load equal to 2 which is due to the lowest priority of the DEFERRED message.

4.4.2 Packet Delay vs. Load

The packet delay vs. the offered load with and without acknowledgment is presented in Figure A.12 for the HIGH priority. The observation is consistent with the earlier ones, namely, the difference increases as the offered load is increased which is due to the increased delay between the time when a packet is ready for transmission to the time when the packet successfully gets into the channel and occupies the channel when IACK is involved in the channel transmission. The saturation of packet delay

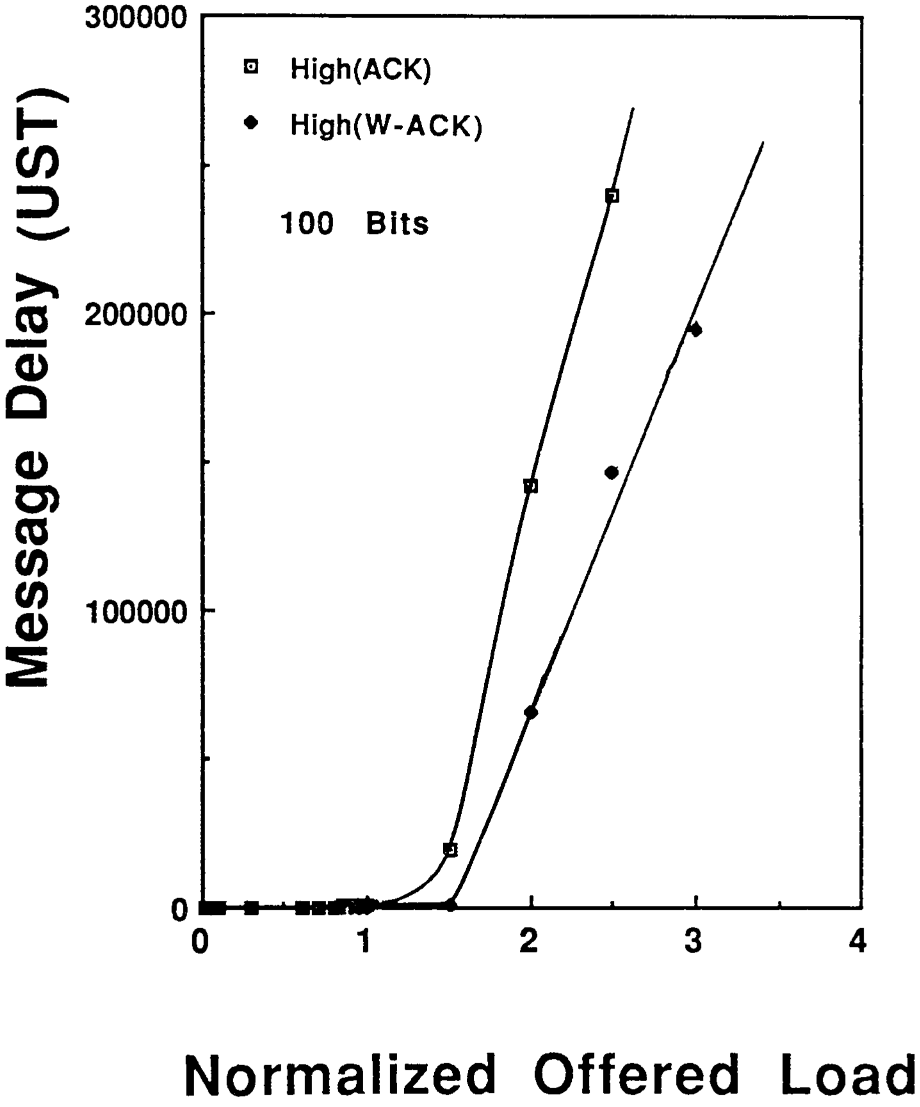

Figure 4.5: Message delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 100-bit HIGH priority message.

appears when the load reaches 2 and 1.6 for unacknowledged and acknowledged cases, respectively, as expected and explained earlier.

The situation is completely different for lower priority packets. Figure A.13 shows the acknowledged and unacknowledged packet delays for the STANDARD priority packets. The packet delay is much larger than that of HIGH priority case and it continues to increase after the offered load is more than 1.0. Here, the delay is not plotted here beyond the load equal to 1.3 and 1.5 for acknowledged and unacknowledged cases, respectively, because it is noticed that the number of STANDARD priority message successfully transmitted through the channel already reduces to below 10%.

For the DEFERRED priority shown in Figure A.14, it is seen that there is a sharp increase when the offered load reaches 0.4. The DEFERRED packet is of lower priority when compared to the STANDARD packet which explains the earlier appearance of the sharp increase.

4.4.3 Message Throughput vs. Load

The message throughput for the HIGH priority is shown in Figure A.15 where it is clearly seen that the throughput starts to decrease when the offered load is equal to around 1.5 which confirms the explanation for the message delay sharp increase when the offered load is equal to 1.5. The difference in the acknowledged and unacknowledged cases is due to the extra occupation of the channel by the IACK frame which does not count as an information message.

The message throughput for the STANDARD priority is shown in Figure A.16 where it is clearly seen that the throughput starts to decrease when the offered load is equal to 0.8 which again confirms the explanation for the message delay sharp increase when the offered load is equal to 0.8.

The message throughput for the DEFERRED priority is shown in Figure A.17 where it is clearly seen that the throughput starts to decrease when the offered load is equal to around 0.4 which also confirms the explanation for the message delay sharp increase when the offered load is equal to 0.4.

4.5 Comparison of Channel Throughput With and Without IACK

Finally, the channel throughputs with and without acknowledgment are all presented in Figure 4.6 for direct comparisons.

The throughput increases for increased message size which has been reported for unacknowledged studies [15,16]. The difference between the acknowledged and unacknowledged throughputs are due to the extra channel occupation time. Notice that IACK frames do not count as information packets.

It is found that the maximum channel throughput is not realized even when the allowed maximum packet size is used as depicted by the top two curves shown in Figure 4.6.

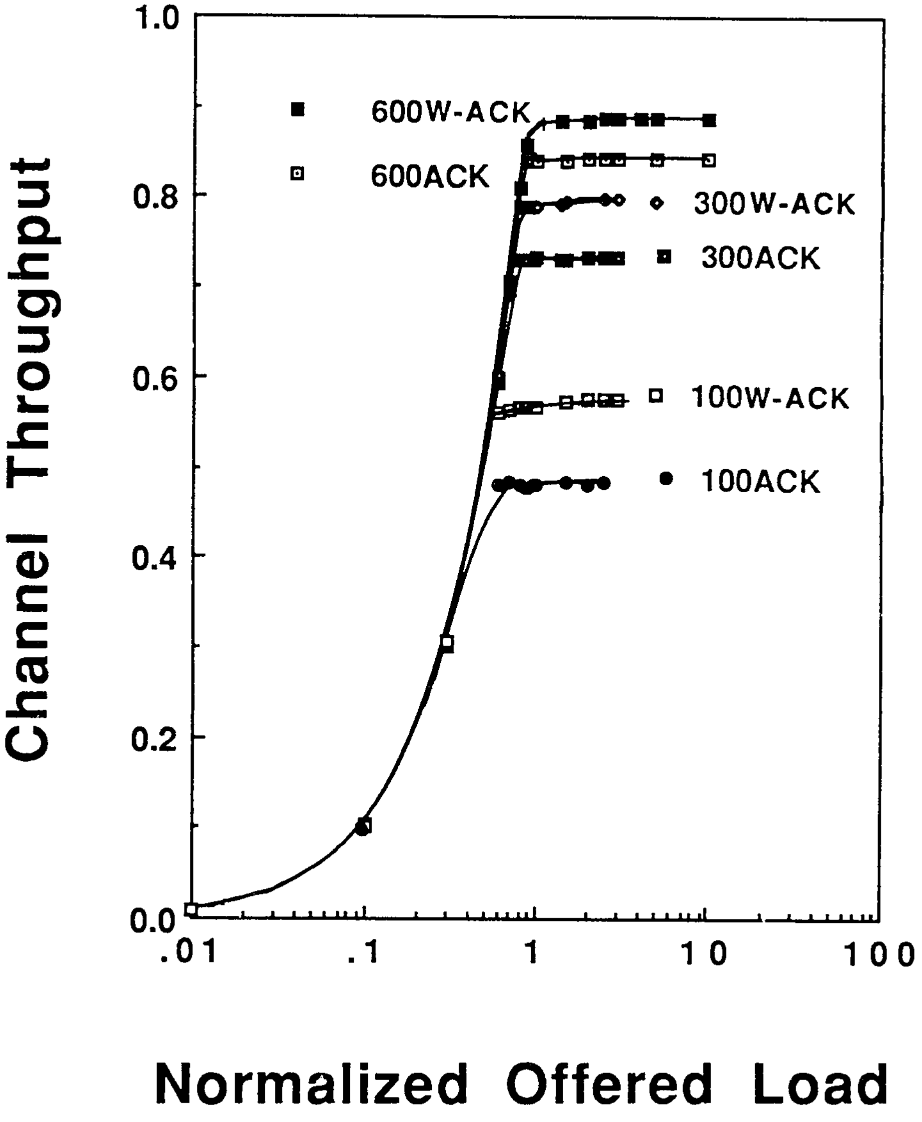

Figure 4.6: Channel throughput vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 600, 300, and 100-bit HIGH priority messages.

CHAPTER 5

CONCLUSIONS

Computer simulation experiments have been carried out to evaluate the performance of the CEBUS implemented using power line. Acknowledged simulation has been done for the first time and a comparison has been made between the performance of acknowledged and unacknowledged cases.

A system of 30 stations (nodes) with 10 for each of the three message priorities, namely, HIGH, STANDARD, and DEFERRED has been investigated. All the simulation results are a statistical average of 5000 messages. Four major performance parameters have been measured, including message delay, packet delay, message throughput, and channel throughput, all as a function of the normalized offered load over a wide range of the normalized offered load. Packet and message lengths vary from around the minimum (100 bits) to near the maximum (600 bits).

Comparisons between the acknowledged and unacknowledged network performances have been made. The acknowledged network using 24 bits contention-free immediate ACK performs very closely to the unacknowledged network when the packet size is around the maximum (600 bits).

Measurable difference between acknowledged and unacknowledged network performances appear when a message size equal to 300 bits is used:

- For message delay, substantial difference starts to appear when the load rises above 2, 1, and 0.7 for the HIGH, STANDARD, and DEFERRED Priorities, respectively.
- For packet delay, difference starts to appear when the load rises above 1, 0.8, and 0.6 for the HIGH, STANDARD, and DEFERRED Priorities, respectively.

- For message throughput, difference starts to appear when the load rises above 2, 1, and 0.7 for the HIGH, STANDARD, and DEFERRED Priorities, respectively, in agreement with the observation for the corresponding message delays.

The performance differences between acknowledged and unacknowledged PL CEBUS network for 100 bits message size include:

- For message delay, substantial difference starts to appear when the load rises above 1.5, 0.8, and 0.3 to 0.4 for the HIGH, STANDARD, and DEFERRED Priorities, respectively.
- For packet delay, difference starts to appear when the load rises above 0.6 to 0.7, 0.4, and 0.3 for the HIGH, STANDARD, and DEFERRED Priorities, respectively.
- For message throughput, difference starts to appear when the load rises above 1.5, 0.8, and 0.4 for the HIGH, STANDARD, and DEFERRED Priorities, respectively, in agreement with the observation for the corresponding message delays.

The maximum allowed packet size of around 600 bits is seen to not have fully utilized the channel throughput. The maximum channel throughput for 600 bits acknowledged and unacknowledged serves are found to be 0.844 and 0.887, respectively. The maximum channel throughput for 300 bits acknowledged and unacknowledged serves are found to be 0.732 and 0.80, respectively. Finally, the maximum channel throughput for 100 bits acknowledged and unacknowledged serves only reach 0.484 and 0.577, respectively.

APPENDIX A

Figures for Chapter 4

This appendix lists the figures discussed in Chapter 4. Four performance parameters have been measured as a function of the normalized offered load. For ease of reference, they are defined again as follows:

1. Packet Delay: it is defined as the time elapsed from the moment a packet becomes ready for transmission in the originating station (node) to the moment the packet is successfully received by a station (node). It does not involve the queueing time.

2. Message Delay: it is measured as the time elapsed from the moment a message is queued at the originating station (node) to the moment the entire message(all its packets) is successfully received by a station (node). Message delay includes the message queueing time.

3. Channel Throughput or Throughput: it is calculated as the ratio of the total number of information bits successfully transmitted through the channel to the simulation time. Note the term "information bit" means a bit belonging to the information part of a packet, i.e. a message bit. Packet headers and trailers do not count.

4. Message Throughput: it is measured as the ratio of the total number of bits received at the destination address to the number of bits generated at the source.

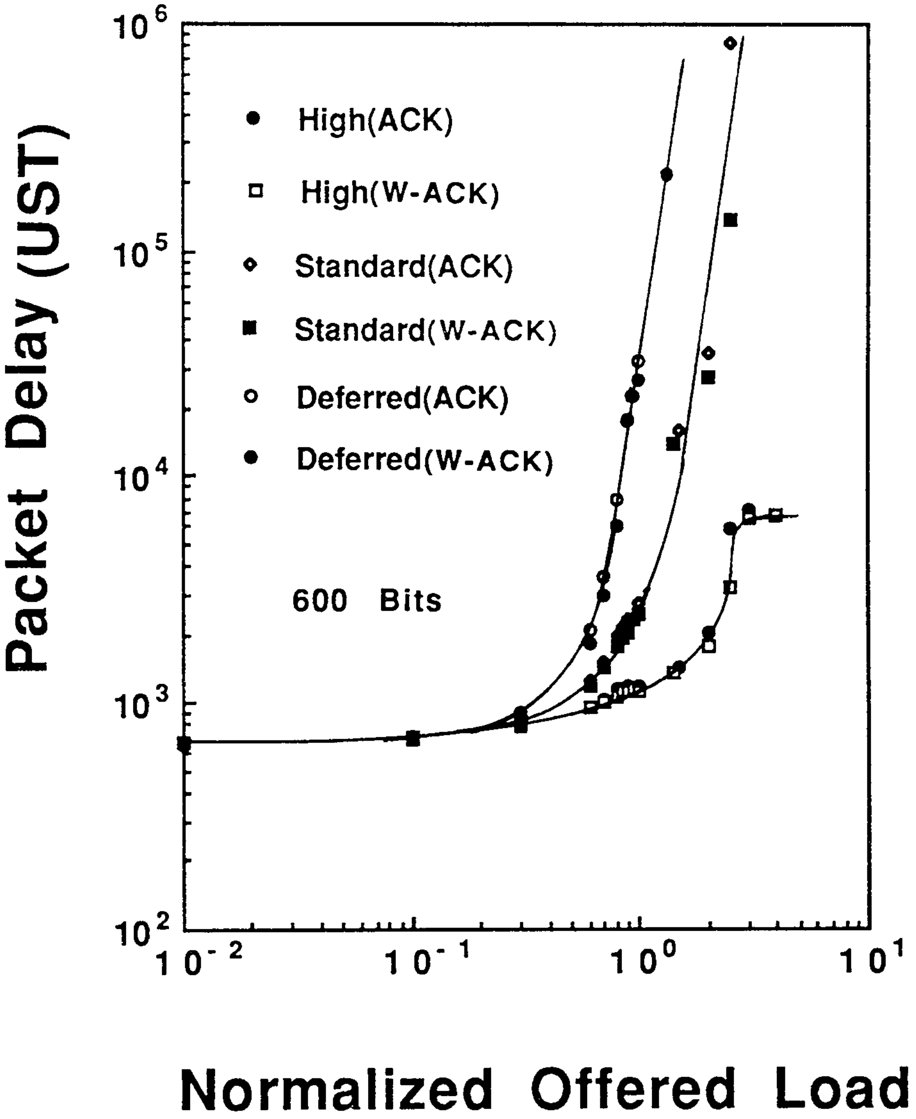

Figure A.1: Packet delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 600-bit packet.

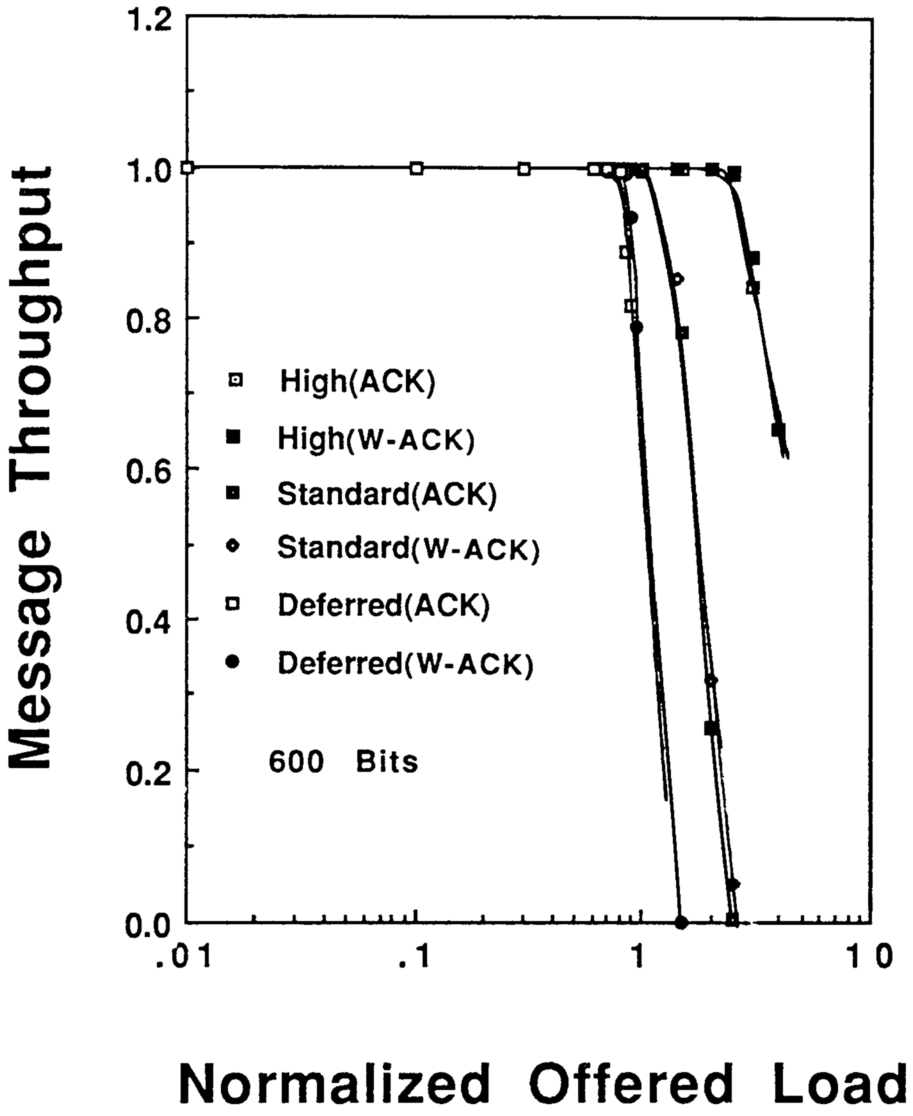

Figure A.2: Message throughput vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 600-bit message.

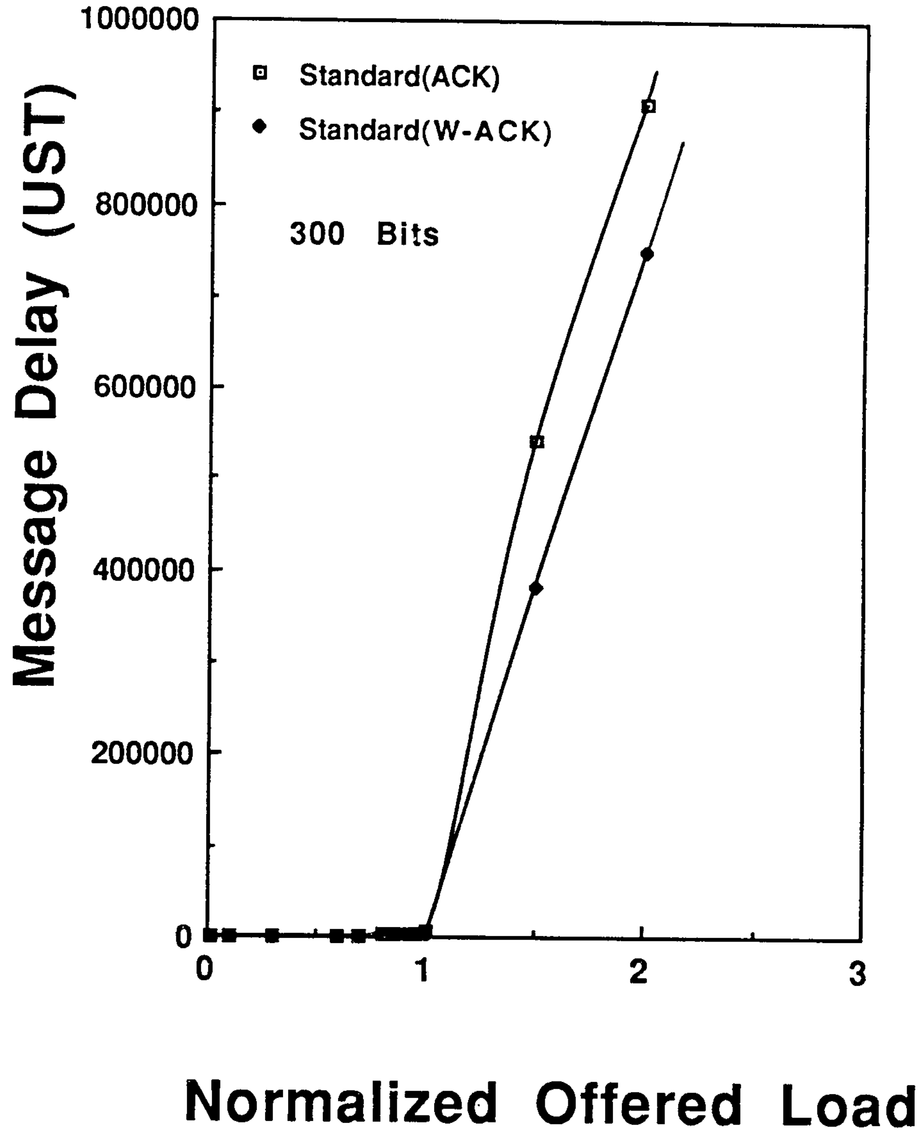

Figure A.3: Message delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 300-bit STANDARD message in linear scale.

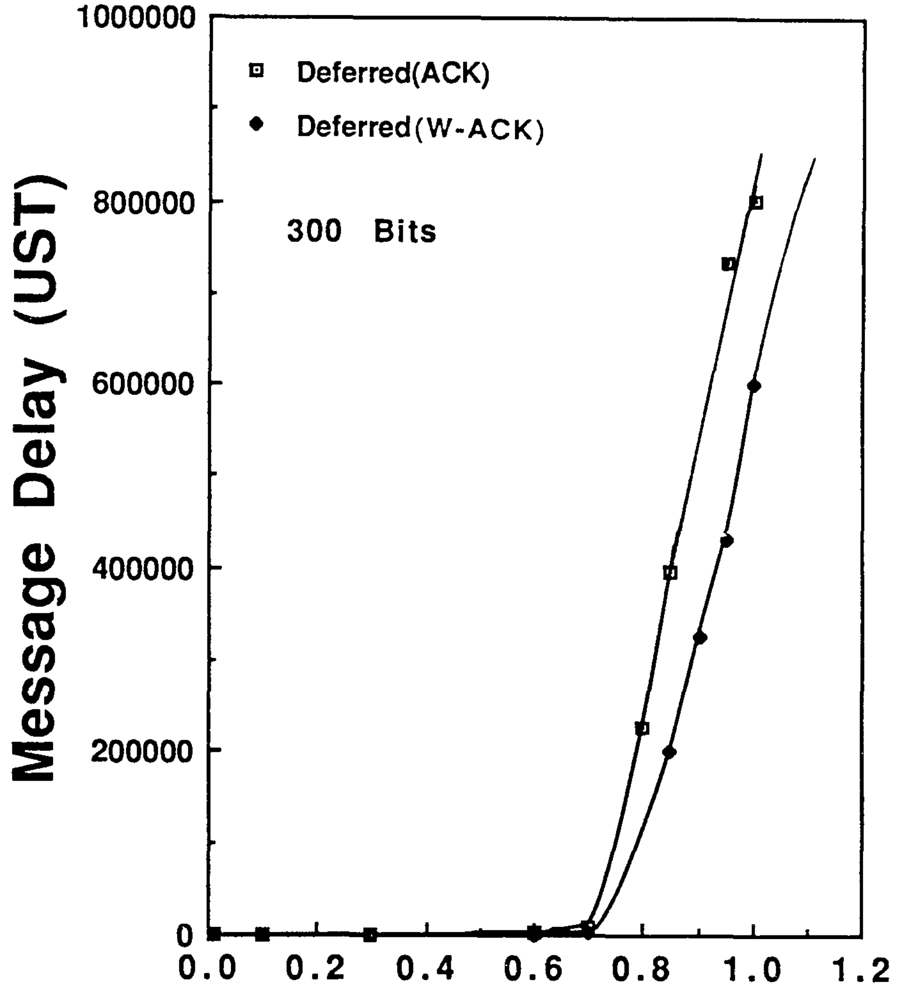

Figure A.4: Message delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 300-bit DEFERRED message in linear scale.

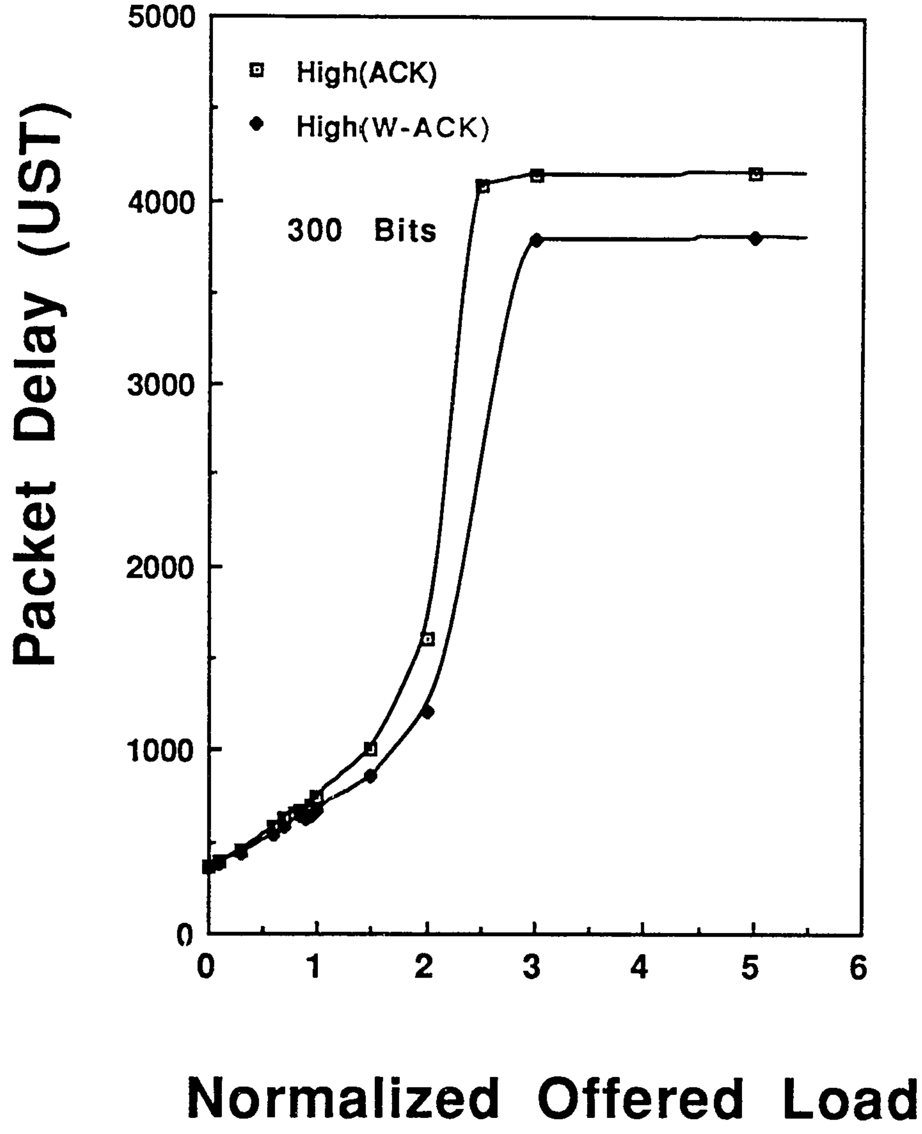

Figure A.5: Packet delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 300-bit HIGH priority packet.

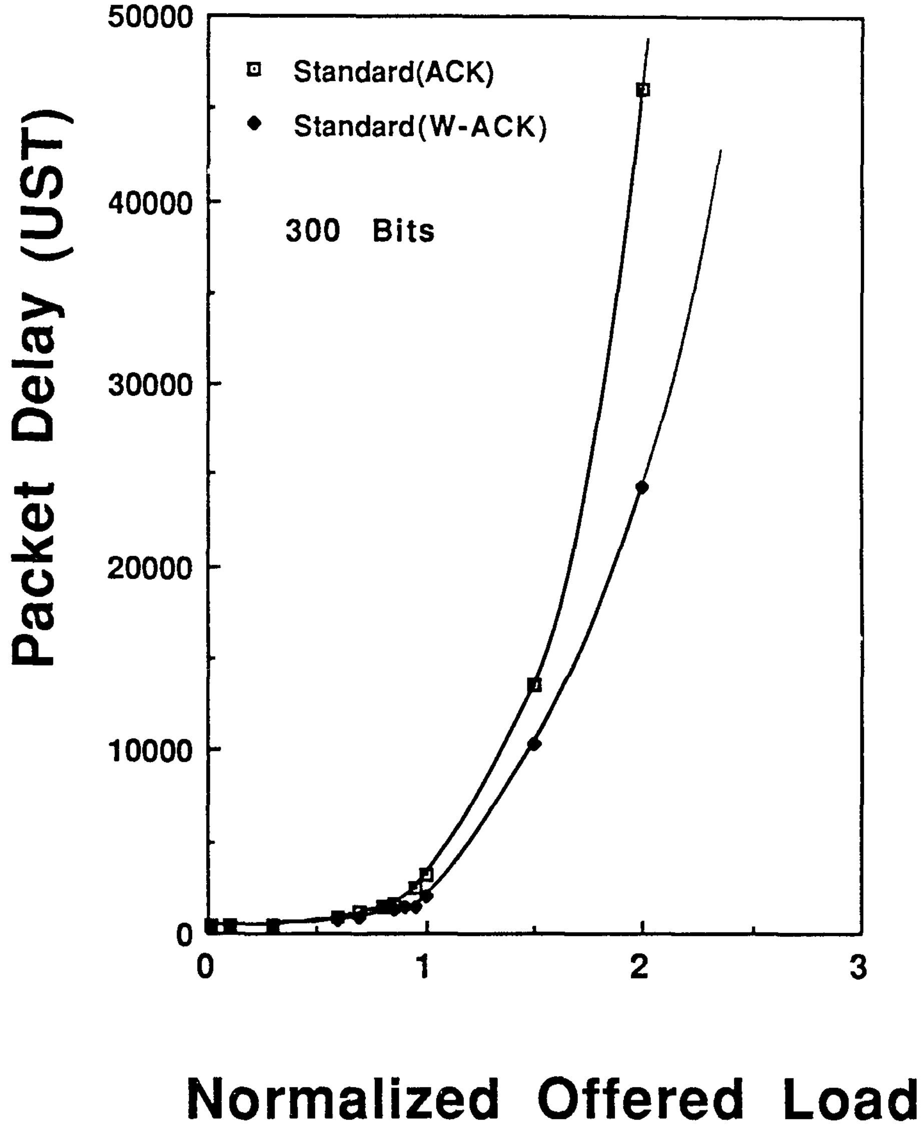

Figure A.6: Packet delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 300-bit STANDARD priority packet.

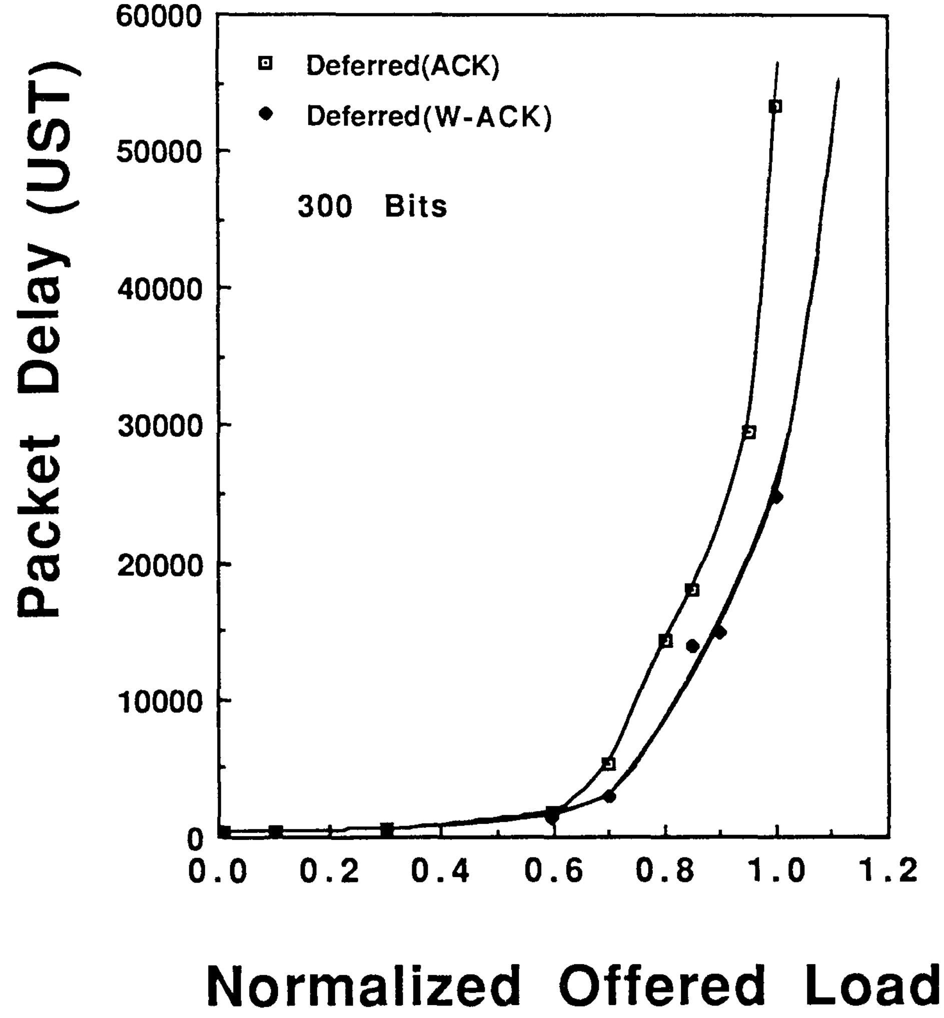

Figure A.7: Packet delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 300-bit DEFERRED priority packet.

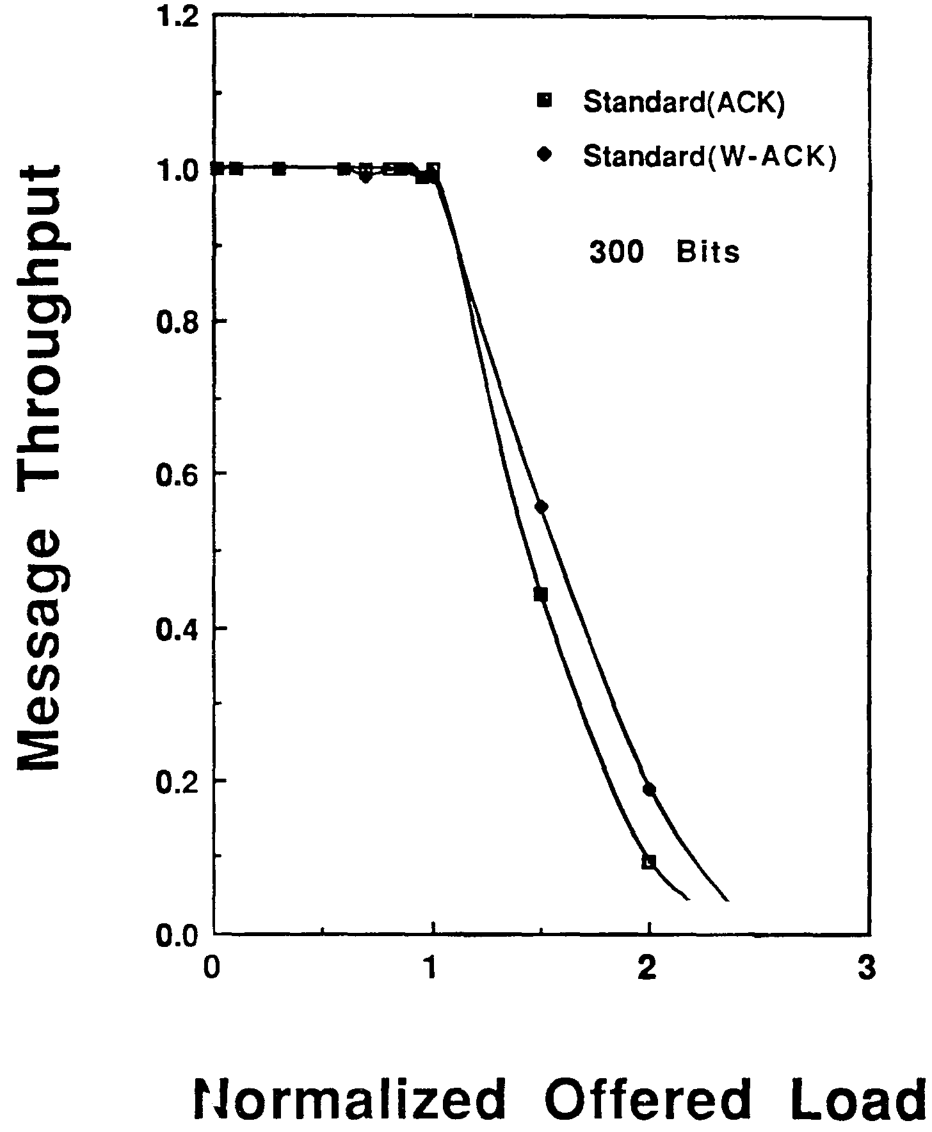

Figure A.8: Message throughput vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 300-bit STANDARD priority message.

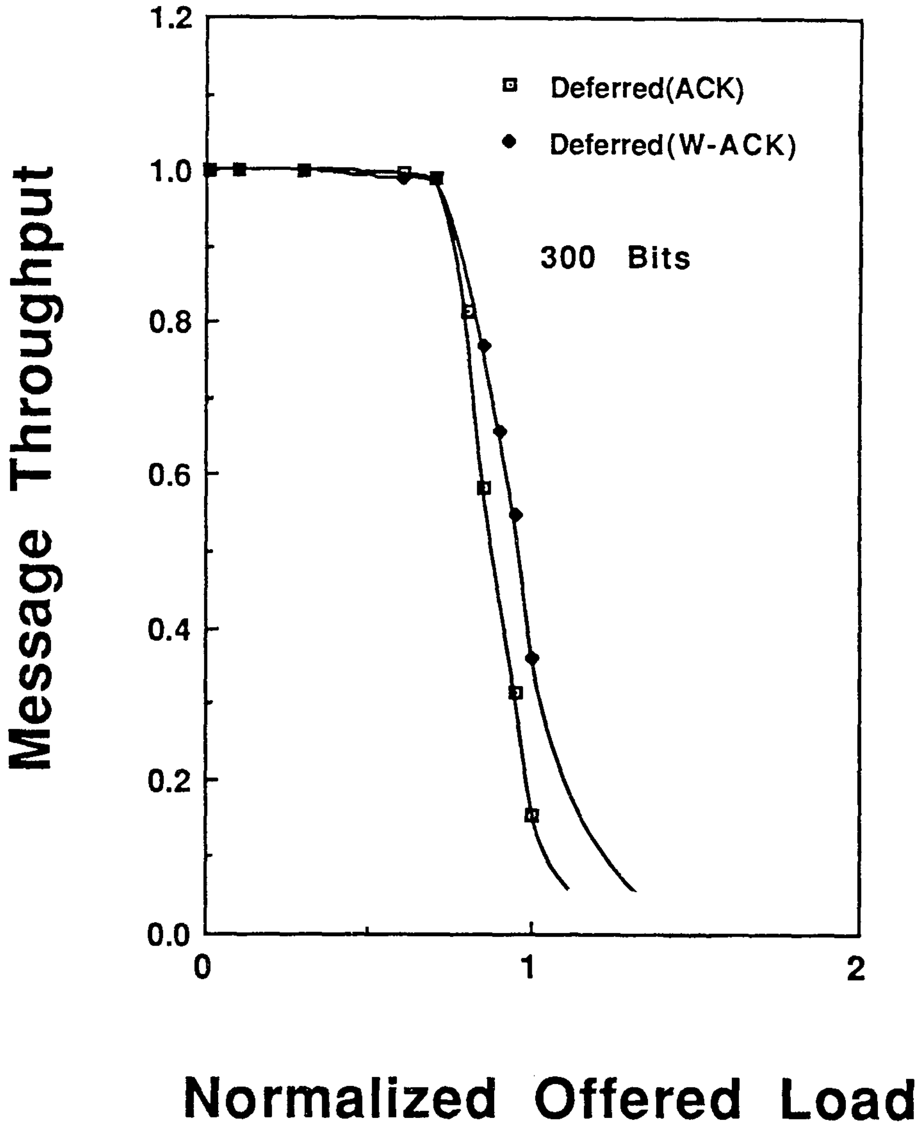

Figure A.9: Message throughput vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 300-bit DEFERRED priority message.

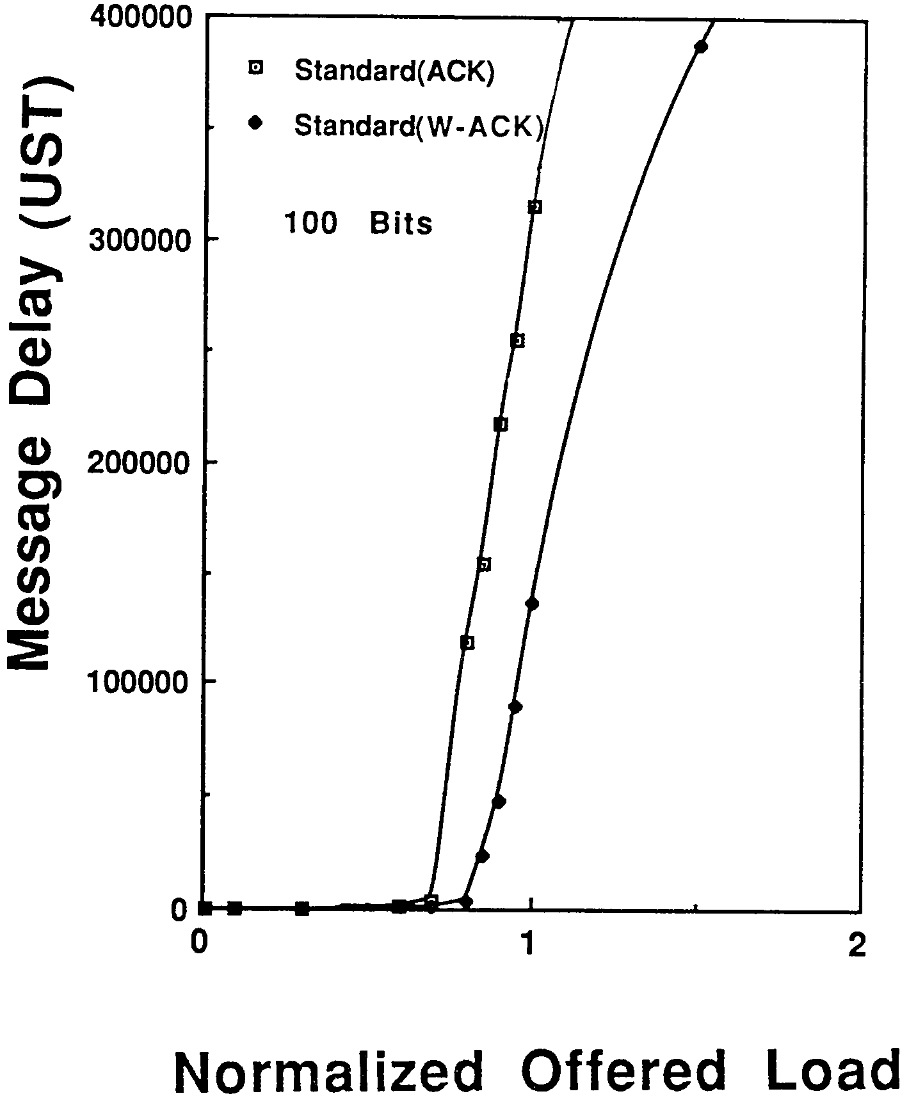

Figure A.10: Message delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 100-bit STANDARD priority message.

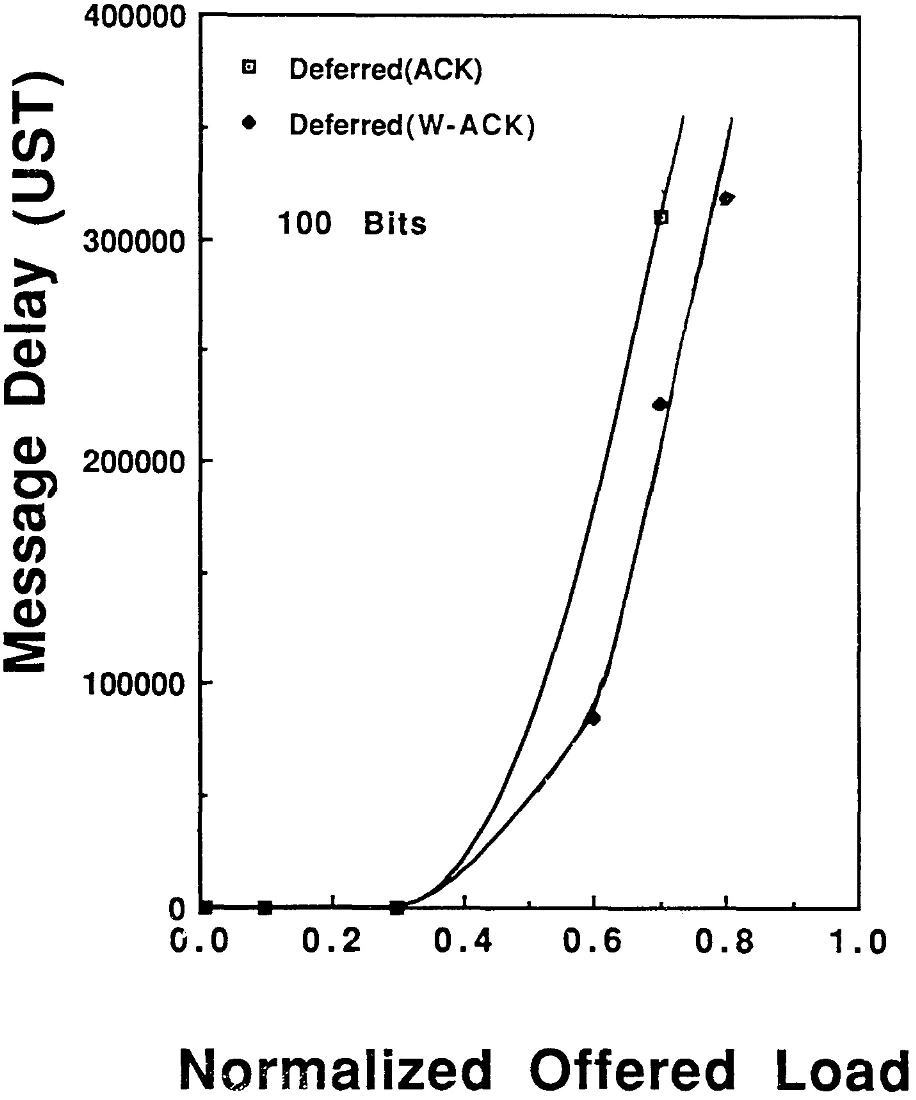

Figure A.11: Message delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 100-bit DEFERRED priority message.

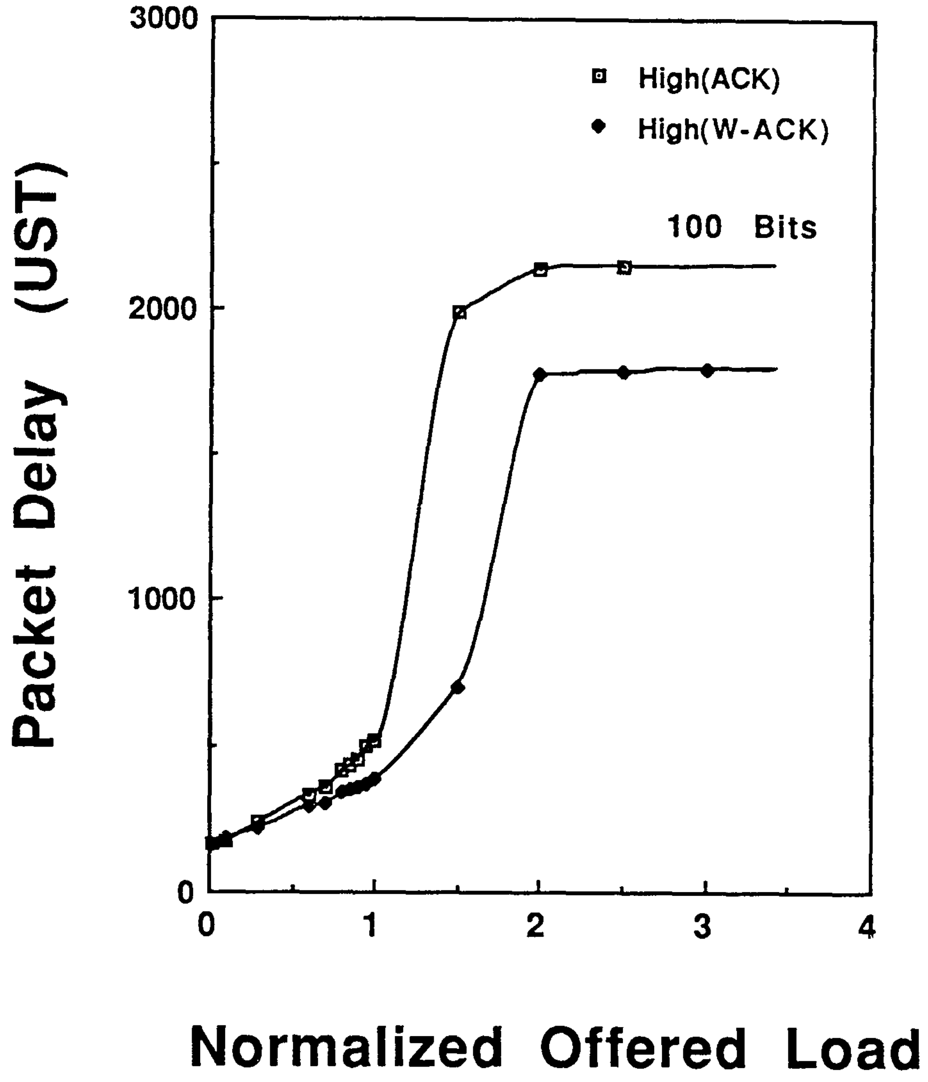

Figure A.12: Packet delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 100-bit HIGH priority message.

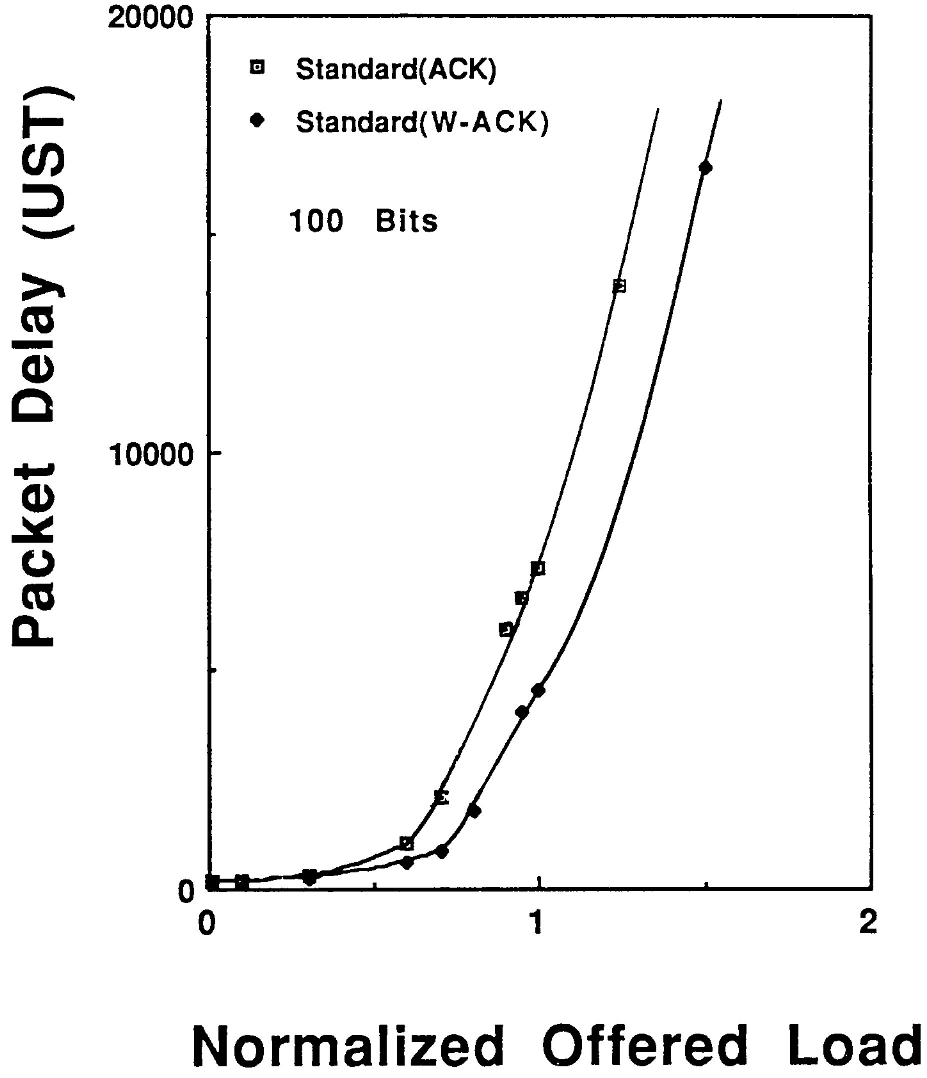

Figure A.13: Packet delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 100-bit STANDARD priority message.

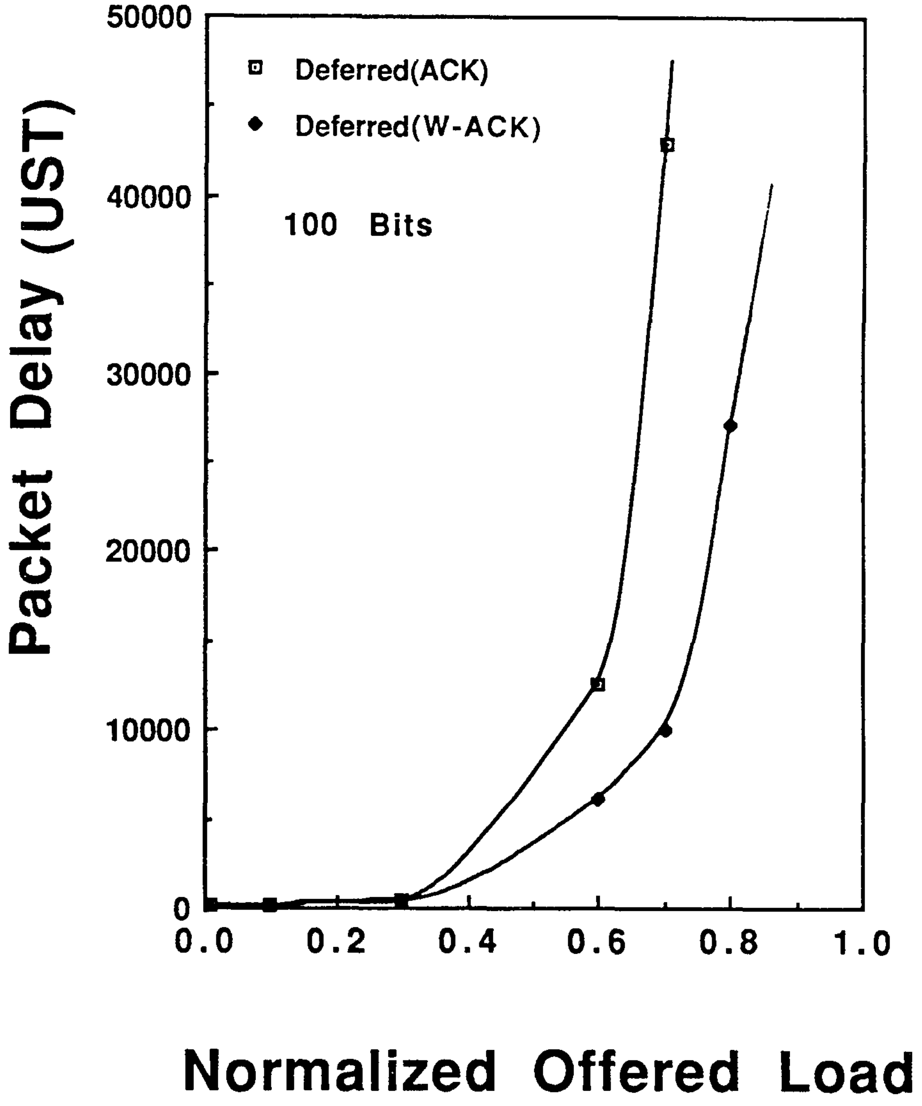

Figure A.14: Packet delay vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 100-bit DEFERRED priority message.

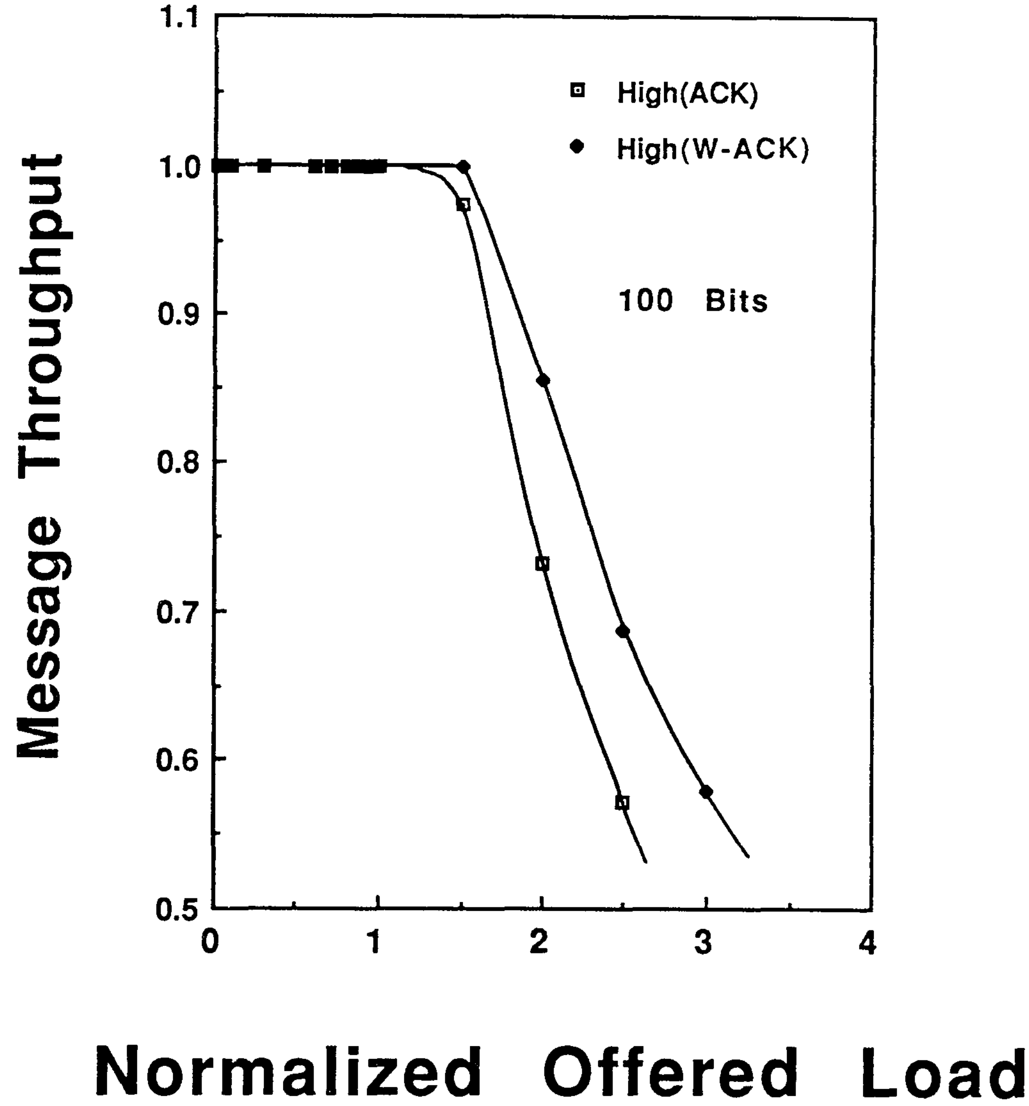

Figure A.15: Message throughput vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 100-bit HIGH priority message.

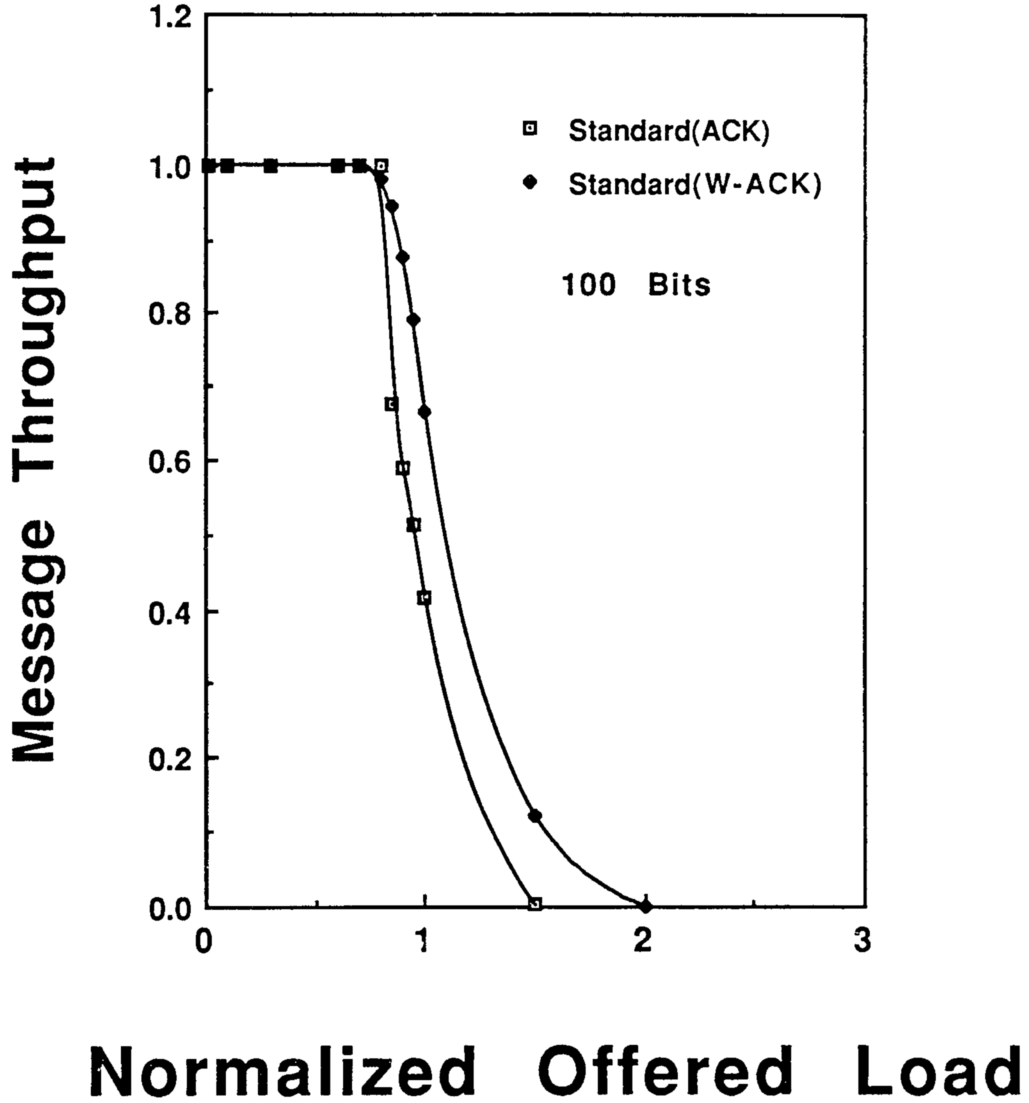

Figure A.16: Message throughput vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 100-bit STANDARD priority message.

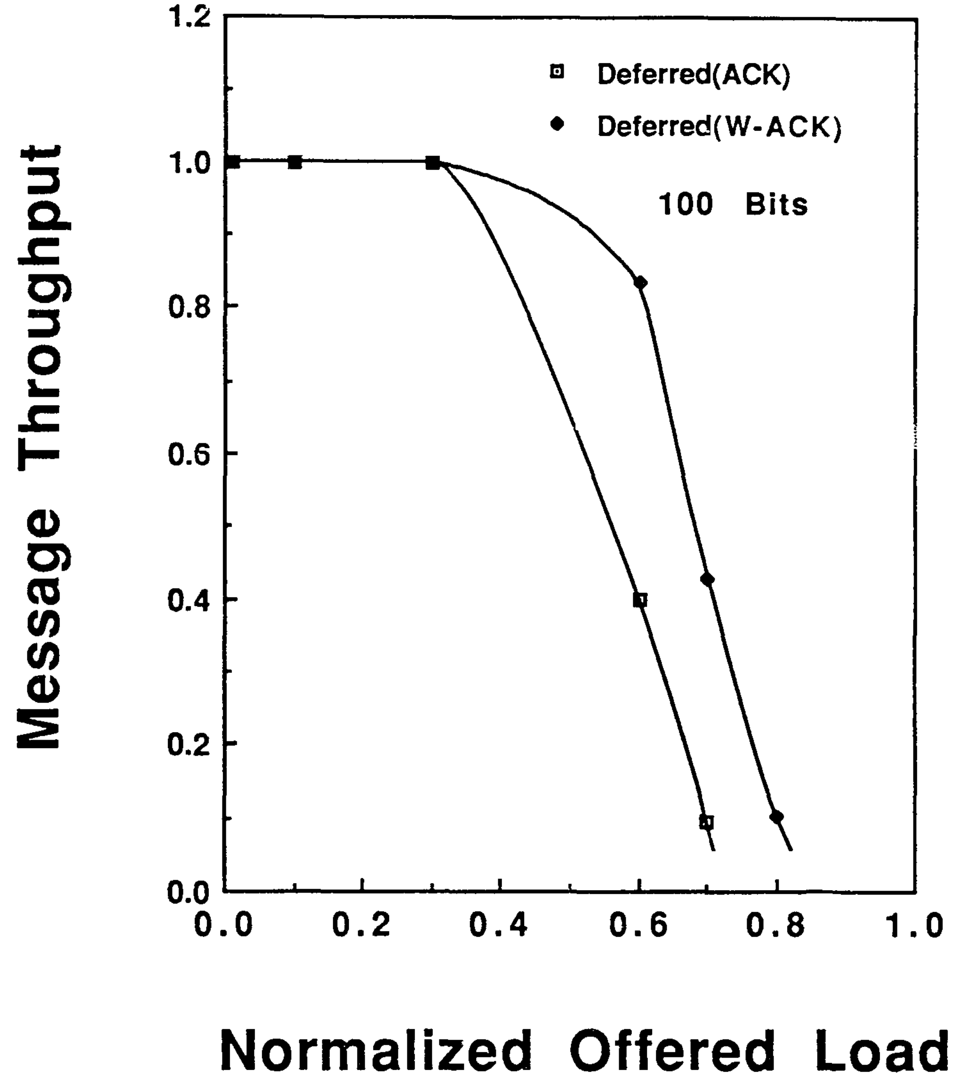

Figure A.17: Message throughput vs. normalized offered load with (ACK) and without (W-ACK) acknowledgment for 100-bit DEFERRED priority message.

REFERENCES

1. *Draft CEBUS PL Specification:* Vol. 1 thru 3, Revised Version, EIA, Oct. 21, 1991.

2. The Electronic Industries Association's, *EIA Interim Standard*, September 25, 1992.

3. *The CEBUS Router Protocol Draft Standards:* Vol. 6 Revised Version, EIA, Jan. 15, 1992.

4. EIA-XXX.1 CEBUS Introduction and EIA-XXX.3 Physical Layer and Media Specification, *Draft CEBUS Introduction and Draft Specification*, Oct.-Nov., 1990.

5. *CEBUS MAC-Layer Specification*, Draft Proposal, July 11, 1988

6. *Node Protocol Draft Standards:* Vol. 4 Revised Version, EIA, Nov. 15, 1991.

7. Tanenbaum, A. Computer Networks, Prentice-Hall, Inc., Englewood Cliffs, New Jersey 1988.

8. Schwartz, M. Telecommunication Networks: Protocols, Modeling and Analysis, Addison-Wesley, 1987.

9. Chlamtac, Imrich, William R. Franta and Dan Levin, "BRAM: The Broadcast Recognizing Access Method," *IEEE Transactions on Communications*, Vol. Com-27, No.8, August 1979, pp. 1183-1189.

10. Signorile, Robert, James LaTourrette, Michael Fleisch, " A study of A Priority for PC Based Local Area Networks Supporting Large File Transfers," *Proceeding, 1987 Symposium on the simulation of Computer Network*, 1987, pp. 123-128.

11. Kiesel, M. Wikhard, and Paul J. Kuehn, "A New CSMA-CD Protocol for Local Area Networks with Dynamic Priorities and Low Collision Probability," *IEEE Journal on Selected Areas in Communications*, Vol.SAC-1, No.5, November 1983, pp. 869-876.

12. Pakkam, S. R. and C. N. Manikopoulos, "Performance Evaluation of the Consumer Electronic Bus". *IEEE Transaction on Consumer Electronics*, Vol. 36, No. 4, Nov. 1990, pp. 949-953.

13. Markwalter, B. E. and S. K. Fitzpatrick, "CEBUS Network Layer Description," *IEEE Transaction on Consumer Electronics*, Vol. 35, No. 3, 1989, pp. 571-576.

14. Gburzynski, Pawel, and Piotr Rudnicki, *The LANSF Protocol Modeling Environment*, Department of Computing Science, The University of Alberta, Edmonton, Alberta, Canada, 1989.

15. Yang, J. and C. N. Manikopoulos, "Router Connected Physical Media in Networking the Intelligent Home", *IEEE Transaction on Consumer Electronics*, Vol. 38, No. 1, Feb. 1992, pp. 30-36.

16. Yang, J. and C. N. Manikopoulos, "Investigation of the Performance of Controlled Router for the CEBUS", *IEEE Transaction on Consumer Electronics*, Vol. 38, No. 4, Nov. 1992.