Performance Analysis of HDLC Protocol Operating in Asynchronous Balanced Mode.

by

Upendra M. Kulkarni

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Electrical Engineering

APPROVED:

Dr. M. A. Wortman, Chairman

Dr. S. F. Midkiff

Dr. J. G. Tront

May 1988

Blacksburg, Virginia

Performance Analysis of HDLC Protocol Operating in Asynchronous Balanced Mode.

by

Upendra M. Kulkarni

Dr. M. A. Wortman, Chairman

Electrical Engineering

(ABSTRACT)

The objective of this work is to analyze the performance of HDLC Balanced Class of Procedures under saturated, full-duplex transmission on error prone links. This thesis extends work done by Bux et al. [8] by considering errors on both the links. Satellite links have long propogation delays compared to terrestrial links, and hence, have longer error recovery times. For such links, errors in acknowledgements have considerable impact on the throughput. In this analysis, the effect of errors in acknowledgements is taken in to consideration. An analytical approach is used to derive performance measures. The concept of "virtual transmission time" introduced by Bux et al. is redefined to accommodate the effect of errors in acknowledgements and used in the analysis. Resulting throughput calculations show how various parameters, (e.g. transmission rate, propagation delay, error rate and packet size), interact and determine the performance.

Acknowledgements

I would like to express my gratitude and respect to my advisors Dr. M. A. Wortman and Dr. T. T. Ha. It has been an honor working under them for this thesis. Additional thanks are due to Dr. Wortman for valuable guidance and encouragement.

Thanks are due to my committee members Dr. S. F. Midkiff and Dr. J. G. Tront for their guidance and encouragement throughout this work.

I would also like to thank Dr. H. F. Van Landingham, Ex-Chairman of the Graduate Committee and , Graduate Consuler, for their cooperation and support during my master's program.

I dedicate my work to my parents and to my wife, , for their love and support which is invaluable.

Table of Contents

List of Illustrations

1.0 Introduction

The data link layer in computer communication networks governs functions such as addressing, frame numbering, error recovery and flow control etc. in order to facilitate reliable data transfer between two computers connected by an error prone communication link. The set of procedures of the data link layer that performs these tasks, known as protocol, determines behavior and performance of the data transfer process. Computer manufacturers first defined such protocols, (e.g., DDCMP- Digital Equipment Digital Data Communication Message Protocol [1], IBM SDLC- IBM Synchronous Data Link Control Procedures [1] etc). International standardization bodies attempted to standardize these protocols in order to provide the computer users and manufacturers with a protocol that suited a wide range of applications. Examples of these are ISO HDLC- High Level Data Link Control, ANSI ADCCP- Advanced Data Communications Control Procedures and X.25- LAPB etc.

There are many protocols in use today. The most popular are the stop-and-wait protocol, the go-back-n protocol and the selective reject protocol. The later two are essentially "sliding window protocols" and are more efficient than the former one. Idealized versions of these protocols have been analyzed for throughput, waiting time etc. In [1,2,3] the stop-and-wait protocol is analyzed using different approaches. Schwartz [2] and Tanenbaum [1] derive expressions for throughput based on the basic operation of the protocol. Towsley and Wolf [3] use a mathematical model of the protocol and discrete time analysis to derive throughput expressions. The go-back-n scheme is analyzed by Konheim in [4] using a discrete time approach.

Konheim also analyzes the selective reject protocol. In [2], Schwartz gives analysis of the go-back-n scheme on terrestrial and satellite links for point to point links. Sabnani and Schwartz further analyze the go-back-n and selective reject protocols on broadcast satellite links [5], and show that the selective reject scheme performs better than the go-back-n scheme on broadcast links. Yu and Majithia [6] analyze the window mechanism of the sliding window protocols. The effect of various window sizes and packet lengths on throughput and waiting time are discussed.

The HDLC protocol has three modes of operation and is suitable for a variety of computer networks. The Asynchronous Balanced Mode (ABM) of operation suits the needs of a point to point link with both ends having equal control over the link. It uses the principle of "sliding window mechanism" [1,7] for the data transfer process and "go-back-n" [1,7] for error recovery. The HDLC protocol provides special features to help speed up error recovery and improve performance. Bux, Kummerle and Truong [8] present a detailed analysis of Balanced Asynchronous Class of Procedures which govern the ABM mode using both an analytical approach and simulation. The model underlying their study has two hosts connected by a full-duplex communication link. The performance measures are derived assuming that only one station transmits data, that the link is saturated in data and acknowledgements are always transmitted successfully. The key idea involved in this analytical approach is the "virtual transmission time" of a frame. They showed that the performance measures can be easily computed using this idea. Also they have presented both analytical and simulation results which further support the theory of virtual transmission time. The work of Bux et al. [8] relaxes many of the assumptions necessery to the analysis of previous work reported in the literature.

Wang J. [9] has extended work of Bux et al. by deriving transfer time and throughput results for unsaturated, variable packet length data networks. Gopal I. [10]

has considered the other modes of HDLC suitable for broadcast links and derived throughput expressions.

The objective of this thesis is to analyze the performance of the HDLC Balanced Class of Procedures under less restrictive assumptions than Bux et al., while using the concept of virtual transmission time. Here we drop the assumption that the acknowledgement frames are always transmitted successfully. This makes both links error prone, and thus, our model reflects the operation of the HDLC even realistically.

The concept of "virtual transmission time"[8] extends easily to our model. We redefine the virtual transmission time of a frame to accommodate the effects of errors in acknowledgement frames. Expressions for performance measures are derived using their redefined virtual transmission time.

2.0 The Data Link Layer

The data link layer, which belongs to level 2 of the ISO (International Standards Organization) OSI (Open Systems Interconnection) network reference model, performs the task of providing the network layer with a "transmission-error free" communication link. It utilizes an unreliable (error-prone) channel provided by the physical layer and transforms it into "error-free" channel by using special data transfer procedures.The messages to be transmitted are broken into "data frames" by the data link layer before transmitting them. To accomplish error free operation, these frames are transmitted sequentially and acknowledgements from the receiver are processed to ensure correct and orderly delivery of the frames. Since the physical layer is concerened only with transmitting simple bits over the channel, it is the responsibility of the data link layer to define frame boundries by attaching special bit patterns at each of a frame. Special care must be exercised to avoid confusion between the frame boundries and similar data pattern that may occur in the frame.

The data link layer software accounts for frames transmitted and acknowledgements received so as to detect those frames damaged during transmission and take necessery corrective actions. Retransmission of a frame is invoked by either of two events; first, the receiver sends a negative acknowledgement requesting retransmission of a frame in error or second, sender gets no response from the receiver. In the later case, the transmitter waits for certain time (known as time-out) and then retransmits the frame. The time-out period must be large enough to allow for two-way propogation delay of the channel and, usually, a small processing time at the receiver. Although retransmission helps recover frames in error, it does introduce an additional problem which must be handled at the data link layer. Consider a situation as follows; the sender transmits a frame and it arrives at the receiver error free. The receiver then computes checksum, accepts the frame, transmits an acknowledgement and waits for next frame to arrive. Suppose this acknowledgement is destroyed, and the sender upon time-out, retransmits the frame. This retransmitted frame also arrives at the receiver without errors and it is accepted as the next frame in sequence. The receiver cannot distinguish between original frames and retransmissions, thus creating a duplicate packet. The receiver then sends an acknowledgement which arrives at the sender safely. The sender now reads next frame from the input buffer and transmits it. The protocol fails due to a duplicate packet at the receiver which went unnoticed. The data link layer must provide for some mechanism to handle such problems in order to provide the network layer with an "error free" channel. Of the several techniques suggested to overcome this problem, we will consider one that is used in HDLC in detail later.

Another problem to be considered at this layer is that of keeping a fast transmitter from "drowning" a relatively slow receiver in data. Further error arise in case of full duplex links; acknowledgement frames compete with data frames for the use of link. We will see how these problems are handled in HDLC as we discuss the operation of the protocol. Finally, the protocol must include procedures to set-up the link for communication and to terminate the link when it is no longer needed or becomes too noisy.

Schwartz [11] gives detailed descriptions of all three phases of data link layer; namely, link establishment, data transfer and termination. We will concentrate on the data transfer phase since our objective is to analyse the data transfer phase of HDLC protocol.

2.1 Protocols

A data link protocol is a set of defined procedures that transform an unreliable communication link into a link that appears error free to the higher level layers. This set of procedures must be capable of handling all the errors considered in the previous section. Several existing protocols, vastly different in principle, resource requirements and implementation, are in use which perform these tasks. We will briefly discuss the three most commonly used in practice, namely, the stop-and wait protocol, the go back n protocol and the selective reject protocol.

The stop-and-wait protocol is the simplest of three. In this scheme only one frame at a time is transmitted. The transmitter then waits for an acknowledgement or a request for retransmission from the receiver. In case of a retransmission request or time-out expiration, the frame is retransmitted. Only when an acknowledgement arrives is the next frame in sequence transmitted. This protocol is well suited for half-duplex links, where two stations alternate transmission. This protocol is ineffecient on full-duplex links with round trip propogation delay much greater than packet transmission time [2].

Most point to point links in long haul networks are full duplex and have a one way propogation delay that is much greater than a frame transmission time. In such cases, it would be a waste of bandwidth for the sender to transmit a frame, wait until an acknowledgement is received, and then transmit next frame. The time sender spends waiting for an acknowledgement is better utilized in the pipelined protocols as explained follows.

The sender transmits up to (M-1) data frames continuously, each with consecutive sequence number (usually 0 to M-2) before blocking transmission. M is chosen such that sender transmits frames without blocking until an acknowledgement for first frame transmitted is due to return. After receiving this acknowledgement, the sender begins transmitting next frame. Now, the sender will receive acknowledgements spaced by one packet transmission time, and hence, will receive permission to transmit next frame when needed. At all times, (M-1) frames are unacknowledged (or outstanding). This situation is equivalent to having a window size of M. This technique is known as "pipelining", and it improves protocol effeciency if the round trip propogation delay is considerably greater than the transmission time of a single packet.

Although it has the advantage of improved effeciency, pipelining frames over an unreliable channel has serious drawbacks since a frame in a sequence may be destroyed. (Recall that the task of the data link layer is to provide the network layer with sequential delivery of frames).

When a frame in the middle of sequence is in error, a large number of succeeding frames will have arrived at the receiver before the sender is aware of the error. The receiver will discard the frame in error, but there may be good frames following the damaged frame. Two schemes are currently in use that account for such a situation [2]. One, called, go-back-n, requires the sender to transmit all the frames following and including the frame in error. The receiver sends a negative acknowledgement for the frame in error and diacards any further frames until it receives the expected frame undamaged. (This corresponds to receive window size of one; at any time only one frame will be accepted.) In case the receiver's negative acknowledgement is damaged or the receiver fails to send an acknowledgement, the sender will time-out on that frame and will retransmit all the frames since the last acknowledgement. Note for high error rates this approach wastes bandwidth.

The other scheme is called selective reject. The receiver will request retransmission of only the frame in error. It will buffer any succeeding frames received until it has all the frames in sequence and then forward them to the network layer. If the sender detects error within a frame only this damaged frame is retransmitted. This approach must specify receiving window greater than one since any frame falling within the window may be accepted and buffered until all preceeding frames have been received and passed to the network layer. Selective reject has the advantage of better utilizing the channel bandwidth in the presence of high error rate, but it requires large memory buffers at receiving end to store frames received out of order.

3.0 Operation of The Protocol

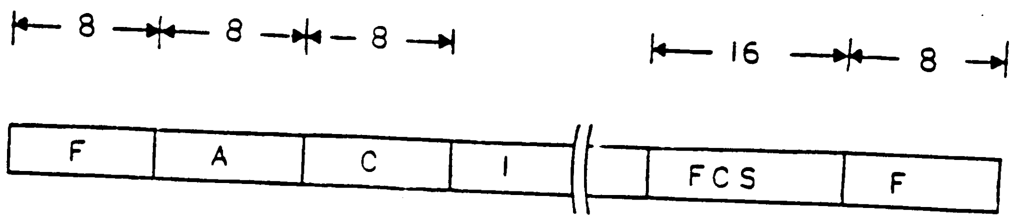
3.1 Introduction

In this chapter, the operation of the protocol is discussed briefly. The definitions of basic terms used in the discussion are considered first. The data transfer phase and error detection and recovery are discussed with illustrative examples later.

The basic unit used for data transmission is called a frame. The standard frame format for HDLC protocol is shown in Figure 1.

The flag field which appears at the beginning and the end of every frame consists of 8 bits and is used to establish and maintain synchronization. Bit sequence 01111110 is reserved for flag and may not appear elsewhere in the frame. If it does then "bit stuffing" is used to achieve data transparency; that is, a 0 is inserted by the transmitter whenever five consecutive 1's are detected in the frame outside the flag field. Thus a string 011111110 would be transmitted as 0111110110. These zeros are removed by the receiver to obtain the original data. If seven 1's appear anywhere in the frame, it is declared in error.

An 8 bit address field is used to identify the receiver or the sender of the frame. The command frames are always sent with receiving station' address and the response frames are always sent with sending station's address. On multidrop lines, the address field identifies one of the terminals in the network. For point to point links, the address field is sometimes used to distinguish command frames from re-



F - Flag

A - Address

C - Control

I — Information

FCS — Frame Check Sequence

Figure 1. HDLC frame format

sponse frames. The control field identifies the function and purpose of the frame. Three types of frames are defined by the control field:

1) Information frame (I)

2) Supervisory frame (S)

3) Unnumbered frame (U)

Figure 2 shows a break-up of the 8 bit control field.

An information frame is identified by a 0 in bit location C1 and conforms to the [F,A,C,I,FCS,F] format as shown in the Figure 1. Only I-frames carry the data field which contains the information to be transmitted. The 3 bit number N(s) identifies the sequence number of the frame. The number N(r), the piggyback ack, acknowledges receipt of all I-frames with number N(r)-1 and less in the reverse direction. Thus, a frame with N(s) = 110 and N(r) = 101 has sequence number = 6 and acknowledges frames 4 and less, indicating that frame number 5 is expected. N(s) and N(r) are modulo 8 for normal HDLC and modulo 128 for the extended version of HDLC. Thus, for normal HDLC, these numbers cycle through the set {0,1,...,7}.

The information field in I-frames carries the data to be transmitted. It can be of any length (since the flag field at the end of the frame serves as the "end of block" marker) and can consist any bit pattern since bit-stuffing is used to achieve data transparency. All I-frames have an overhead of 48 bits which compose the flag, address, control and frame check sequence fields.

A supervisory frame is identified by bits 10 in locations C1,C2 shown in Figure 1; there are four types of S-frames. The N(r) field in S-frames can be used to acknowledge frames travelling in the other direction on the link. More details on the types and use of S-frames are presented later. The U-frames are identified by 11 in locations C1,C2 and are used to set-up the path for data transmission and later on clearing it. There are 32 types of U-frames and are discussed in detail in [2, 12]. Both

	C1	C2	C3	C4	C5	C6	C7	C8
I-Frame	0		N (S)		P/F		N (R)	
S-Frame	0	0	S	S	P/F		N (R)	
U-Frame	1	1	M	M	P/F	M	M	M

Figure 2. Control field

the frames carry only supervisory and control information and conform to [F,A,C,FCS,F] format as shown in Figure 1.

The P/F (poll/final) bit in all three frames (I-frame, S-frame and U-frame) is used for check-pointing purposes which are explained later. The bit is considered to be P in command frames and F in response frames. A command frame with P = 1 invokes a response frame with F = 1.

The 16 bit FCS (frame check sequence) field contains CRC (Cyclic Redundancy Check) information for error detection purpose. There are various methods to carry out the error detection and correction process. One way is to add sufficient redundancy to original data so that errors can be detected and corrected. The number of check bits required for both detection and correction is much larger than that required for detection only [13]. In data networks, typical packet size varies from 1000 to 5000 bits/packet. It is usually more economical to add only enough information to detect errors and have the frame in error retransmitted if it is in error. Tanenbaum [1] describes techniques to compute the check bits (checksum), and describes their utilization in error detection. The transmitting station computes the checksum and appends it to the data field. The CCITT V.41 generator polynomial [1, 2] ($1 + x^5 + x^{12} + x^{16}$) is used to generate this information. All frames are checked for errors in transmission by the receiving station and frames failing to pass FCS check are discarded. In such case, the transmitting station is required to retransmit the frames in error.

3.2 Procedure Classes

Three modes of operation are defined for the HDLC protocol:

1) Normal Response Mode (NRM) :

In this mode one station acts as the master or primary station of the network. All data transfers and responses are initiated by this station. The network may have multiple secondary stations.

2) Asynchronous Response Mode (ARM) :

This mode of operation is similar to the NRM except that a secondary station can initiate data transfer.

3) Asynchronous Balanced Mode (ABM) :

This mode suits the point to point transmission needs of two stations in equal control of the link connecting them.

First consider the various control frames and their use in the link establishment and the data transfer phase.

In HDLC, U-frames are used for setting-up and disconnecting the link. The extended operation (modulo 128) is also specified in U-frames. Once the link is set-up, I and S-frames are used for data transfer, error detection and recovery operations. The control field in I-frames indicates the frame number, P/F check-pointing and acknowledgement for previously received frames. The S-frame can be used to signal a positive acknowledgement using N(r) field in it in absence of an I-frame. The S-frames have four formats which are as follows:

SS	Name	Function
00	RR (ready to receive)	A positive ack for all frames upto and including N(r)-1
01	REJ (Reject)	Rejects all frames N(r) and onwards; acks N(r)-1 and previous frames.
10	RNR (Not ready to receive)	Temporarily blocks the data flow and acks N(r)-1 and previous frames.
11	SREJ (Selective reject)	Request retransmission of N(r).

A RR (ready to receive) frame positively acknowledges all frames upto and including N(r)-1. Such an acknowledgement may be used in absence of an I-frame to enhance the speed of data transfer. On the other hand, in order to control data flow, RNR (not ready to receive) may be used to block data flow when desired. Data transfer may be resumed by the blocking station issuing an RR command. The remaining two S-frames are used in error recovery.

An REJ frame rejects N(r) and all the frames that follow. The transmitting station must retransmit all these frames. Here, the ABM mode uses the go-back-n feature explained previously. The SREJ command is used to request retransmission of a frame specified by N(r). This command is employed in the NRM and ARM modes.

All frames have a P/F (poll/final) bit which enables the command-response mechanism to be carried out. A 1 in this bit in a command frame is defined to be P. This P bit must be responded with a 1, defined to be F, in a response frame. In ABM, P/F can be used to force the other station to send an immediate acknowledgement (S-frame) by setting P = 1 in an I or S-frame. In NRM it is used to poll a set of terminals. The master station sends a frame with P = 1 to the station it wants to poll. The receiving station may respond with a series of frames with the last one having F = 1.

3.3 Asynchronous Balanced Mode

The operation of HDLC under ABM mode is governed by the balanced asynchronous class of procedures (BAC) with predefined functions and options. The functions include use of I, RR and RNR frames. The options are use of REJ (option 2) so as to enhance the speed of error recovery, and the restriction that I-frames be commands only (option 8). The composite class of procedures is called BAC2,8. U-frames, used to set-up and disconnect the link, are also included in the predefined functions of BAC2,8.

We will now see the basic operation of the ABM mode involving data transfer, error handling and recovery. First, consider some notations adapted to describe frames as shown in Figure 3. The nomenclature is self-explanatory.

The configuration of the HDLC balanced class data link procedure is shown in Figure 4.

Two combined stations (transmitter/receiver) are connected by a full-duplex link as shown in Figure 4. The data link layer software for these stations follow the HDLC protocol balanced class of procedures. Both the stations are capable of transmitting and receiving the data once the link has been established. To establish the link, the station with data to be sent sends a connect-request (U-frame) to the other station. The other station may reply either with a connect-confirm or a connect-clear command depending on its priorities. If the other station replies with a connect-confirm command, the link is said to be "set-up". The data transfer can now proceed. Whenever a station has data to be transmitted, it constructs a frame (I or S) and transmits it. After all the data have been exchanged, the link is disconnected by one of the stations upon consultation with the other.

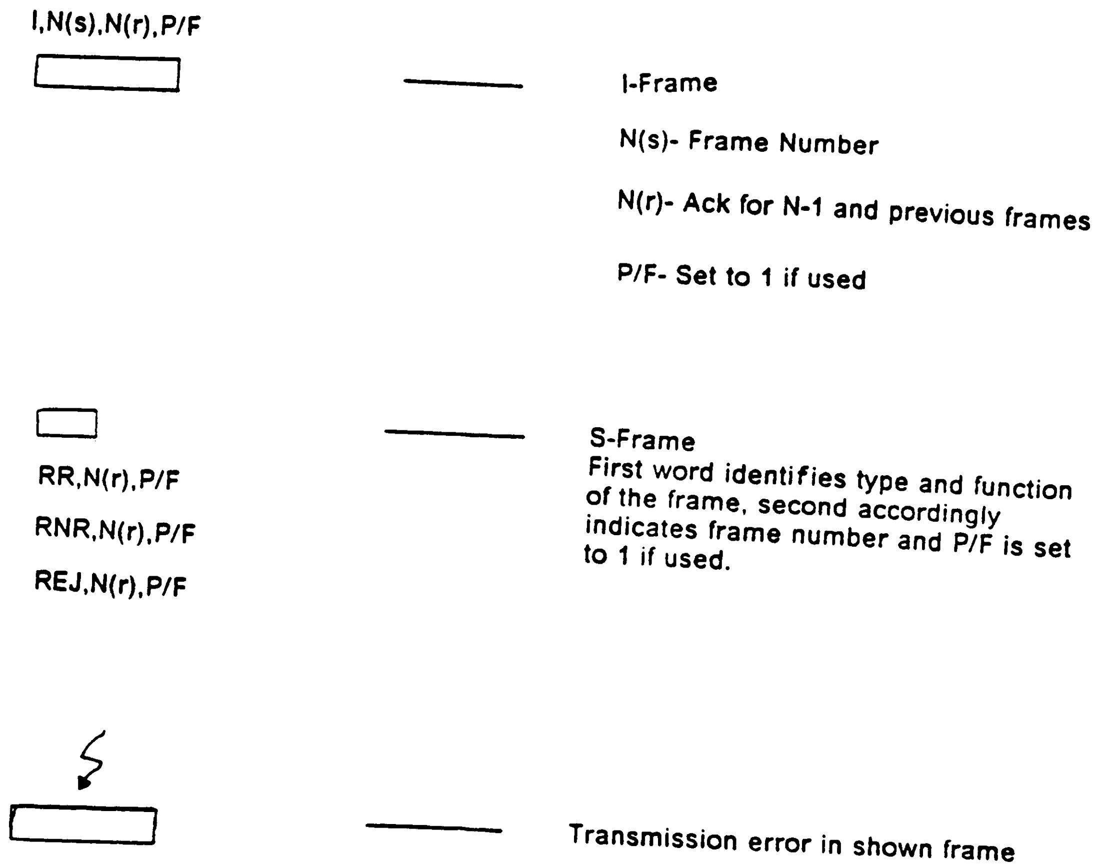


Figure 3. HDLC Notations

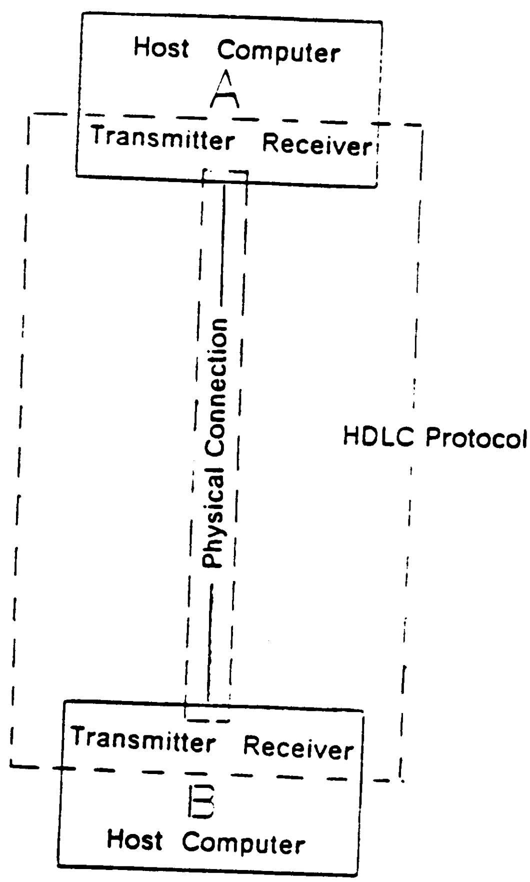


Figure 4. Configuration of a point to point link

To understand the operation of the protocol, we follow an example flow of I and S-frames between the two stations. Figure 5 shows data transfer between two combined stations.

Station A initiates transmission to station B by sending an I-frame numbered BI00. Since it has not yet received any frames from B, A indicates that frame 0 is expected. Station B also begins transmission after a small delay and sends frame AI00 to A. Note that B indicates that frame 0 is expected. At time t1, B has finished transmitting frame AI11, frame number 1. B indicates that frame 1 from station A is expected (since it has received frame 0), and station A has sent frame BI21; frame no. 2 and expected frame from B = 1. Station A then sends BRR1P; bit P = 1, which is immediately responded by station B with BRR3F; bit F = 1. Note here that all command frames carry the address of the destination station, and response frames carry the address of the source station. An S-frame can be either a command or a response. Here BRR3F is a response frame; hence, it carries the address of station B, the source station.

Consider now, how transmission errors are handled and recovery performed. It is assumed that only station A transmits data, and station B transmits acks upon receipt of a frame from station A. Also station A is saturated with data; that is, data packets are always awaiting transmission. The modulus of sequence numbers (M) is assumed to be four. Figure 6 depicts two modes of error recovery; one via REJ command, and the other via timeout recovery.

Station A begins by transmitting three frames (0, 1 and 2) in succession. It must now wait for an ack from station B since the transmit window (size = M-1) is full. The frame 0 reaches station B, and B transmits an ack RR1 indicating frame 1 is expected. Suppose frame 1 is destroyed due to noise on the channel, and is discarded by station B. Frame 2 is received successfully, but since station B is still expecting frame

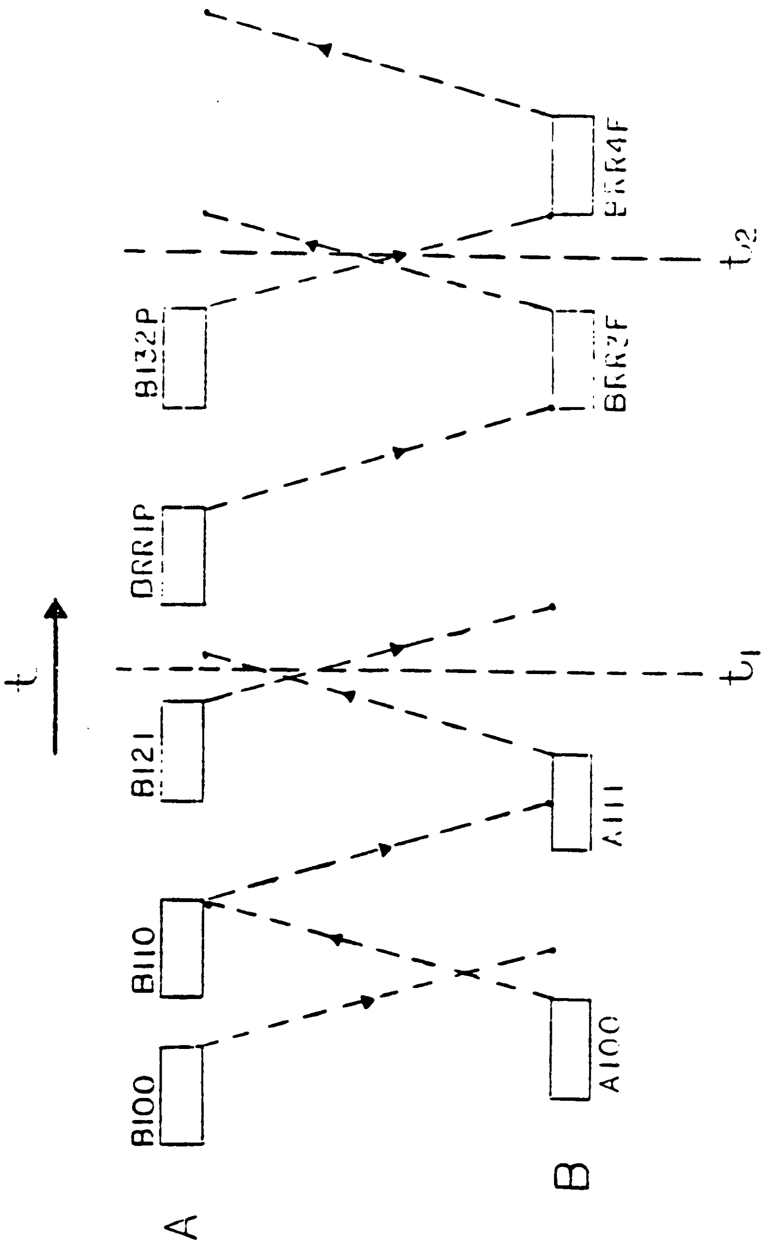


Figure 5. Protocol Operation

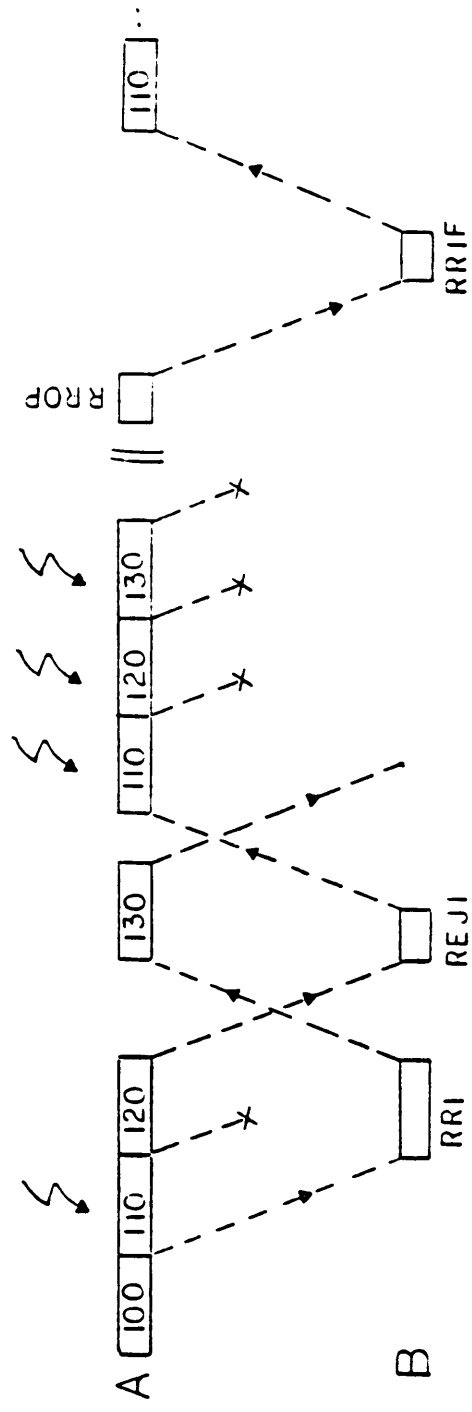


Figure 6. Error detection and recovery

1, it sends REJ1 frame. This indicates to station A that all frames numbered 1 and higher must be retransmitted. Station A, upon receiving this frame, retransmits frame 1 and all frames following. Thus, errors in transmission are detected, and the sending station is made to retransmit the frame in error. This type of recovery is known as recovery by REJ. Note that REJ recovery is be performed since at least one frame following the damaged frame was accepted, and its ack was received before station A timed-out on the frame. An REJ frame may be issued only once and any further errors must be recovered by the timeout mechanism.

Now on retransmission frame BI10 is disturbed once again. The frames BI20 and BI30 are also disturbed. Station B continues to wait for frame number 1. B does not send any response upon receiving garbled frames. Station A eventually times-out on frame BI10, and sends an enquiry BRR0P to station B. The P bit forces station B to send an immediate response. B responds with BRR1F, indicating that it is still waiting for frame number 1. Upon receipt of this response, station A retransmits all the frames following and including frame BI10. This type of recovery is known as time-out recovery. It is clear that the time-out mechanism is mandatory for the recovery process since a control frame may be destroyed due to noise and cannot be recovered by the REJ mechanism. Further, a single I-frame or the last in a sequence of I-frames can not be recovered by REJ. In such cases the time-out mechanism must be envoked.

4.0 Throughput Analysis

In this chapter we derive expressions for throughput of the data transfer phase of the HDLC protocol. The throughput is defined as the number of information bits transmitted successfully per unit time. The principle of virtual transmission time introduced by Bux et al. is used in this analysis. The virtual transmission time is redefined to accommodate effects the assumptions made in this analysis, which are as follows:

4.1 Assumptions

1) Both stations are saturated, that is, information frames are queued for transmission.

2) Both stations are operating under ABM (Asynchronous Balanced Mode) class of operations, but it is assumed that station B sends an information frame with piggybacked acknowledgement only upon receipt of an I-frame from station A. This assumption is made in order to simplify the operation and to obtain an estimate of maximum throughput.

3) REJ recovery is performed only once. Any further attempts to recover a packet are through time-out recovery.

4) Both channels are error-prone with bit error probability P_{Bit} . Block error probability P_B for l bits/block is

$$P_B = 1 - (1 - P_{Bit})^{(l+48)} \qquad [4.1]$$

5) It is assumed that the S-frames are transmitted successfully since they are only 48 bits.

4.2 Virtual Transmission Time

Virtual transmission time t_v of an I-frame is defined as the average length of time required to transmit the I-frame successfully. Virtual transmission time may not be contiguous.

Virtual transmission time for a frame with N(S) = i begins simultaneously with transmission provided that the frame with N(s) = (i-1) was received error free at the destination; ends simultaneously with the end of transmission of the frame, provided the frame is received successfully, and its acknowledgement returns successfully. In case of an error in the acknowledgement frame, the time by which an I-frame is delayed (which would have been transmitted upon receipt of this acknowledgement frame) is counted in the t_v of the frame with N(s) = i. If the I-frame with N(s) = i cannot be transmitted due to (M-1) unacknowledged frames (provided no error occurs in acknowledgement of any of these frames) its t_v is extended by the time it must wait until transmission begins again.

Since t_v is the average length of time required to transmit a packet successfully, throughput is defined as

$$Throughput = I/t_v \qquad [4.2]$$

where I = number of information bits in an I-frame.

Consider the definitions of various time periods involved in the data transfer phase. Figure 7 shows a typical flow of frames between two stations. The channel from station A to station B is refered as 'forward channel' and that from station B to station A as 'reverse channel'. Although both of the stations transmit I-frames and S-frames, the figure only shows information of interest in the analysis such as I0 and

I1 on the forward channel, and RR1 and RR2 on the reverse channel, as shown in the figure.

It is assumed that all messages are of constant length I, so all I-frames are of length (I+48) (where S-frames are 48 only bits). The channel capacity is t_{rate} bits/sec and one way propagation time is t_p. This gives I-frame transmission time t_I as

$$t_I = (I + 48)/t_{trate\ sec} \qquad [4.3]$$

and S-frame transmission time is $t_s = 48/t_{rate}$ sec. It is also assumed that a constant time t_{proc} is required to process a received frame. The acknowledgement time t_{ack} is given as

$$t_{ack} = 2t_p + t_I + t_{proc} \qquad [4.4]$$

In order to determine the average virtual transmission time, there are two different cases distinguished by the relation among transmission time of an I-frame t_I, acknowledgement time t_{ack} and modulus M of the sequence numbers: In case I) the acknowledgement time t_{ack} is greater than the time to transmit (M-2) I-frames, in case II) the time t_{ack} is not greater than (M-2) t_I.

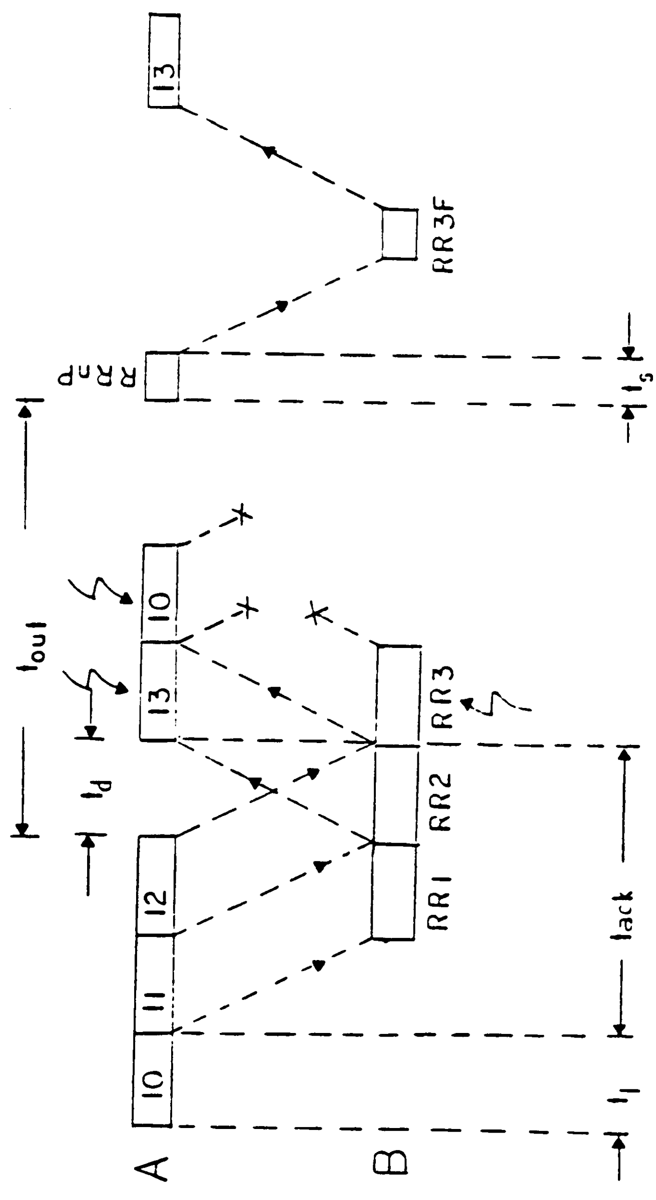


Figure 7. Typical frame transfer sequence

4.3 Case I)

Here, the transmitter exhausts the transmit window of (M-1) frames before the acknowledgement for the first I-frame transmitted returns. There is a delay of t_d before this acknowledgement arrives so that transmission can resume. This delay t_d plays an important role in the data transfer and error detection and recovery phase, and thus, affects throughput. We use following approach suggested by Bux et al. [1], to study the impact of t_d on performance.

A random variable 'W', known as the window width, is defined at the beginning of each virtual transmission time and corresponds to the number of I-frames which station A can transmit before it must stop because of (M-1) unacknowledged I-frames. Thus, W can take on any value in the set of w;

w = { 0, 1, 2, ..., M-2 }

The probability P_w that W takes a value w, and the virtual transmission time $t_v(w)$ for an I-frame for which W equals w must be determined. The mean virtual transmission time can be written as

$$t_v = \sum_{w=0}^{M-2} P_w t_v(w)$$

To determine P_w, we study the window width process as shown in Figure 8.

Figure 8 shows the state transition diagram of the window width process which forms a semi-Markov process. The explanation for the window width process is as follows:

The window width as defined previously can take values from 0, 1, ..., (M-2) but not (M-1) since at the beginning of a virtual transmission time, the preceding I-frame

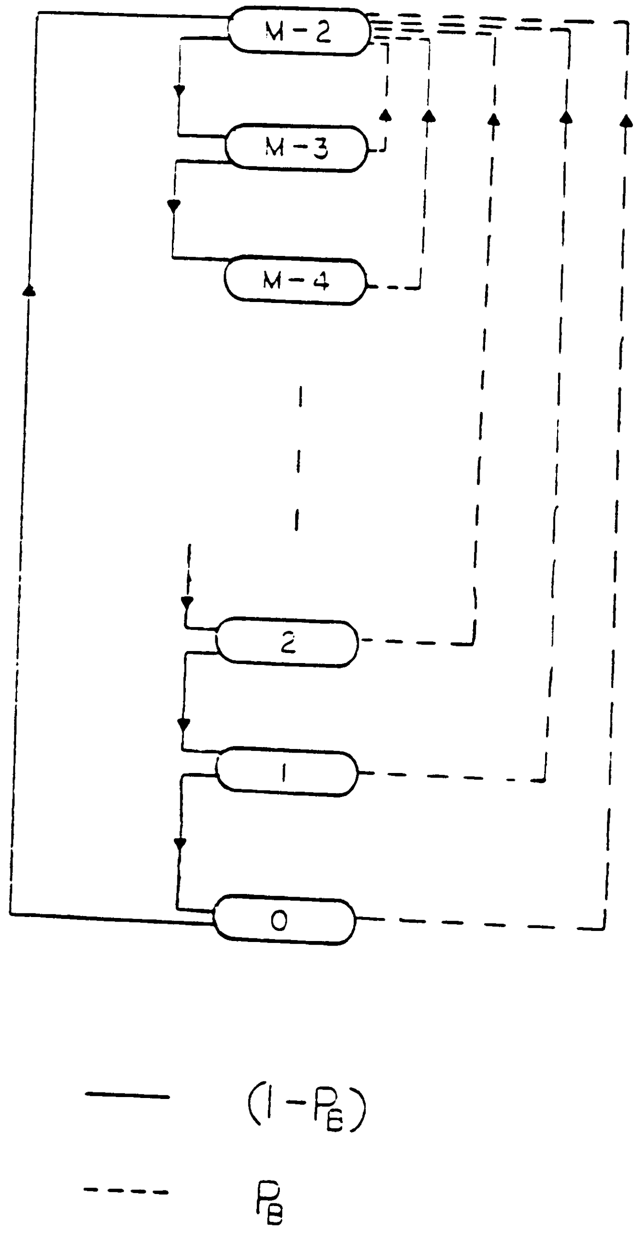


Figure 8. State transition diagram of the window width process

cannot yet have been acknowledged. The number of I-frames that can be transmitted after and including this frame is (M-2) (which is the maximum for W). The window width starts at (M-2) and is reduced by one each time an I-frame is successfully transmitted (probability = 1 - P_B). The minimum value 0 is reached when all but one I-frame in a window have been successfully transmitted. After transmitting an I-frame with w=0, the window width opens to become (M-2) again. I-frames with w=0 see a delay in transmission due to M-1 unacknowledged frames, denoted by t_d. This delay is given by

$$t_d = t_{ack} - (M - 2)t_I \qquad [4.5]$$

Now, consider the effect of errors on the window width process. An error in transmission is to be recovered by "go-back-n" action which implies that all I-frames following and including the frame in error must be retransmitted. The frame in error becomes the first I-frame to be transmitted, followed by (M-3) I-frames before transmission must stop. Thus an error in forward channel means the window width process is discarded and restarts at (M-2). With probability P_B the system returns to state (M-2) from each of the other states. An error in acknowledgement for an I-frame does not require that I-frame to be retransmitted since that frame has been successfully received and the receiver is now expecting next I-frame. This I-frame will be eventually acknowledged by either acknowledgements for following I-frames or by a response to the sender's enquiry (P/F checkpointing) for the frame upon time-out. So, an error in acknowledgement never causes an I-frame to be retransmitted, and hence, does not disturb the window width process. However, ack error does cause a delay in transmission of further I-frames, this delay is counted in the virtual transmission time of the I-frame. The transition probabilities of the window width process depend only upon error probabilities of forward channel. The transition matrix, T_r, for the embedded Markov chain of this process is,

$$
\begin{array}{c}
0 \\ 1 \\ 2 \\ \cdot \\ \cdot \\ \cdot \\ M-4 \\ M-3 \\ M-2
\end{array}
\begin{bmatrix}
0 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 1 \\
X & 0 & 0 & \cdot & \cdot & \cdot & \cdot & 0 & Y \\
0 & X & 0 & & & & & \cdot & Y \\
\vdots & 0 & X & \ddots & & & & \vdots & \vdots \\
\vdots & & 0 & \ddots & \ddots & & & \vdots & \vdots \\
\vdots & & & \ddots & \ddots & \ddots & & \vdots & \vdots \\
\vdots & & & & \ddots & \ddots & \ddots & \vdots & \vdots \\
0 & & & & & \cdot & X & 0 & Y \\
0 & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & X & Y
\end{bmatrix}
\qquad
\begin{array}{l}
X = (1 - P_B) \\[1em]
Y = P_B
\end{array}
$$

The stationary state probability vector P, where

$$P = [P_1, P_2, \ldots, P_{(M-2)}]$$

is the solution of

$$P = PT_r \tag{4.6}$$

where

$$\sum_{i=0}^{M-2} P_i = 1$$

Solving the matrix equation 3.6, we have the following simultaneous equations

$$P_0 = (1 - P_B)P_1$$

$$P_1 = (1 - P_B)P_2$$

$$\vdots$$

$$P_{M-3} = (1 - P_B)P_{M-2}$$

$$P_{M-2} = P_0 + P_B(\sum_{i=1}^{M-2} P_i)$$

$$= P_0 + P_B(1 - P_0)$$

$$= P_0(1 - P_B) + P_B$$

Now the above simultaneous equations are recursive, and it follows that

$$P_0 = (1 - P_B)^{M-2} P_{M-2}$$

Substituting value of P_{M-2}, we get

$$\begin{aligned} P_0 &= (1 - P_B)^{M-2}(P_0(1 - P_B) + P_B) \\ &= P_0(1 - P_B)^{M-1} + P_B(1 - P_B)^{M-2} \end{aligned} \qquad [4.7]$$

which gives P_0 as

$$P_0 = (P_B(1 - P_B)^{M-2})/(1 - (1 - P_B)^{M-1}) \qquad [4.8]$$

Now, using following relation we get other components of vector P.

$$P_i = (P_0)/((1 - P_B)^i) \qquad i = \{1, 2, ..., M - 2\} \qquad [4.9]$$

Thus for w, such that $0 \leq w \leq M\text{-}2$, we have

$$P_w = (P_B(1 - P_B)^{M-2-w})/(1 - (1 - P_B)^{M-1}) \qquad [4.10]$$

Now we shall derive expressions for $t_v(w)$ where $0 \leq w \leq M\text{-}2$. To derive an expression for virtual transmission time, we first decompose it into various time periods as shown in Figure 9.

Figure 9 shows an example frame transfer sequence with occurance of errors. An important point to be noted here is that an error in forward channel requires a frame to be retransmitted whereas an error in reverse channel does not. So they have different effects on the virtual transmission time.

The mean virtual transmission time can be separated into two parts: average time required to transmit a frame successfully on the forward channel, then average time spent waiting for acknowledgement of that frame. In Figure 9, T_0, T_1 and T_2 correspond to the former part and T_3 corresponds to the later part. From Figure 9, when a frame with window size w requires n retransmissions, its virtual transmission time $T_v(w)$ is given by

$$T_v(w) = T_0(w) + T_1(w) + (n-1)T_2 + T_3(w) \qquad [4.11]$$

Now we find the expected value of $T_v(w)$ as

$$\begin{aligned} t_v(w) = E(T_v(w)) = T_0(w) + P_B E(T_1(w)) \\ + (P_B)^2 \sum_{x=0}^{\infty} (x+1) P_B^x (1-P_B) T_2 \\ + \sum_{x=0}^{\infty} P_B^x (1-P_B) P_B E(T_3(w)) \end{aligned} \qquad [4.12]$$

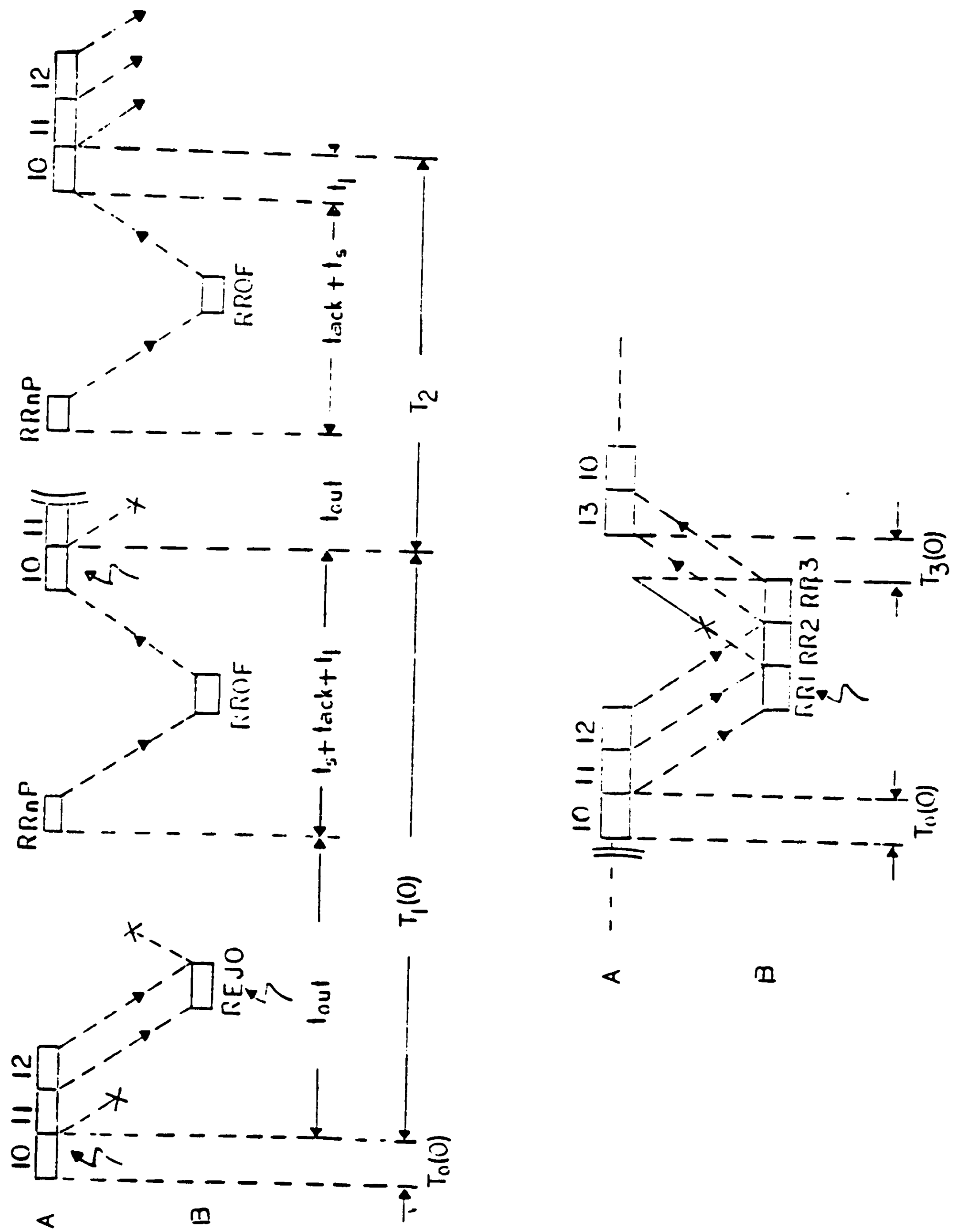


Figure 9. Decomposition of Virtual transmission time

We will consider all time periods that make up virtual transmission time of a frame and derive expressions for each.

Time period T_0 :

T_0 is defined as the time required to transmit a frame for the first time. It is equal to the frame transmission time and depends on the window size as follows

$$T_0(0) = t_l + t_d \qquad [4.13]$$

And for w = {1, 2, ..., M-2},

$$T_0(w) = t_l \qquad [4.14]$$

The window with w=0 sees a delay of t_d before a frame can be transmitted and hence it is counted in $T_0(0)$.

Time period T_1 :

T_1 is counted from the end of first transmission to the end of first retransmission of an I-frame. T_1 depends on the actual window size w and whether or not the recovery is performed via REJ or time-out mechanism.

Recovery via REJ mechanism is performed if and only if not all (M-2) I-frames following the one under consideration are disturbed and the acknowledgement frame for the first I-frame received successfully is received at station A successfully. If this acknowledgement (actually this will be a REJn) is not successfully received, then a time-out error will occur since station B will not respond to any following I-frames. This is so because of the assumption that the REJ mechanism is invoked only once.

First, the expression for $T_0(0)$ will be derived, and then for $T_0(w)$ will be derived taking into consideration both modes of recovery.

REJ RECOVERY

Figure 10 shows a case of recovery via the REJ mechanism. From the Figure 10 it is simple to compute the total probability of this event and its sojourn time as follows:
Total probability P is given by

$$P = \sum_{x=0}^{M-3} P_B^x (1 - P_B)(1 - P_B) \qquad [4.15]$$

This expression corresponds to probability that x I-frames (x < = M-3) are disturbed, the next I-frame is received, and its acknowledgement is received.
The sojourn time T is,

$$T = (x + 1)t_l + t_{ack} + t_l \qquad [4.16]$$

TIME-OUT RECOVERY

Figure 11 shows the two cases of time-out recovery. In first case, all I-frames following the one under consideration are disturbed, and in the other the acknowledgement for first I-frame received at B is disturbed. The time-out mechanism is invoked in both the cases. The total probability P is given by

$$P = P_B^{M-2} + \sum_{x=0}^{M-3} P_B^x (1 - P_B) P_B \qquad [4.17]$$

and the sojourn time T is

$$T = t_{out} + t_s + t_{ack} + t_l \qquad [4.18]$$

Thus, the expression for $E(T_1(0))$ is

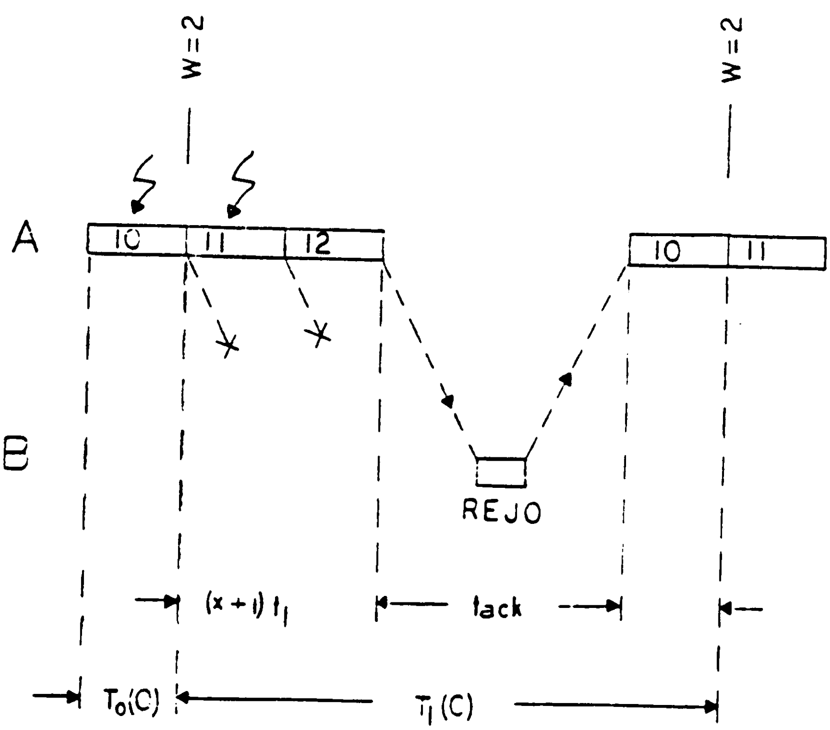


Figure 10. REJ recovery

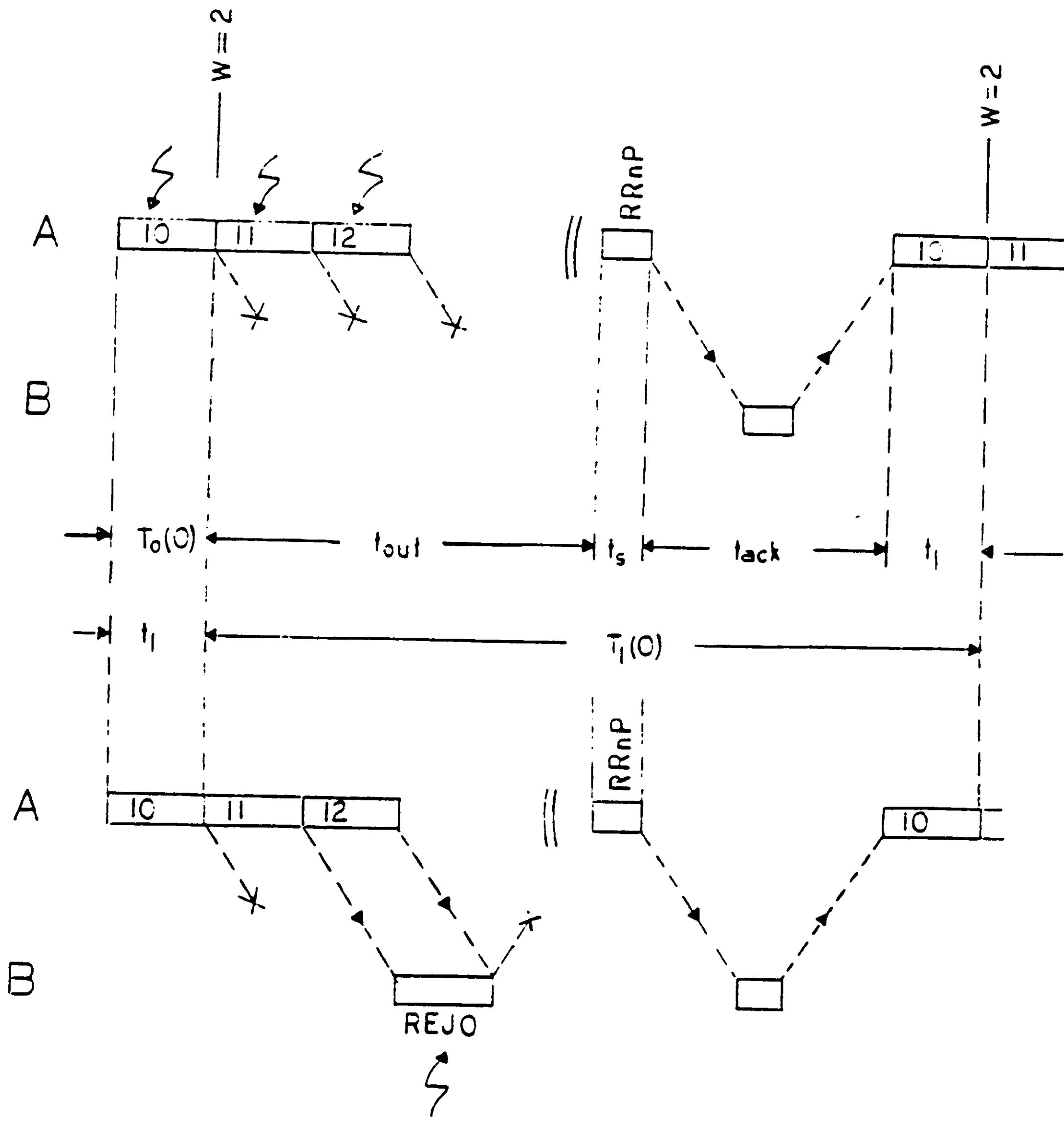


Figure 11. Time-out recovery

$$E(T_1(0)) = \sum_{x=0}^{M-3} P_B^x(1-P_B)^2(x+1)t_I$$
$$+ (P_B^{M-2} + \sum_{x=0}^{M-3} P_B^x(1-P_B)P_B)(t_{out}+t_s) + t_{ack} + t_I \qquad [4.19]$$

This expression is valid for window size w=0. For w= {1, 2, ..., M-2} the expression for $T_1(w)$ is different. Exactly (M-2) I-frames are transmitted after any frame before the transmission terminates due to (M-1) unacknowledged I-frames. In case of w=0, all (M-2) I-frames are transmitted before the delay t_d occurs, whereas for w= {1, 2, ..., M-2}, (w-1) frames are transmitted before delay t_d and remaining (M-2)-(w-1) frames after delay t_d. We must consider time delay t_d in total time spent if all (w-1) I-frames following the one under consideration are disturbed provided at least one from the remaining (M-2)-(w-1) I-frames is transmitted successfully and its acknowledgement is received. Recovery is performed via the REJ mechanism in such case. The probability P of this event occuring is

$$P = P_B^{w-1}(1 - P_B^{((M-2)-(w-1))})(1-P_B) \qquad [4.20]$$

and additional time spent is t_d.

The time-out recovery is performed under similar conditions to the $T_1(0)$ case. The sojourn times and the probabilities of occurance of these events are same. Thus, the expected value $E(T_1(w))$ for w= {1, 2, ..., M-2} is given by

$$E(T_1(w)) = E(T_1(0)) + P_B^{w-1}(1 - P_B^{((M-2)-(w-1))})(1-P_B)t_d \qquad [4.21]$$

Time period T_2 :

T_2 represents time spent to recover frames in error on retransmission. These frames are always recovered by time-out since the REJ mechanism may be used only on first transmission. T_2 is independent of actual window size and is given by

$$T_2(w) = t_{out} + t_s + t_{ack} + t_l \qquad [4.22]$$

Time period T_3 :

T_3 is defined as the time for which transmission of an I-frame (which is being delayed due to (M-1) outstanding acknowledgements) is delayed due to an error in acknowledgement for a previously transmitted frame. T_3 is included in the virtual transmission time of the frame, the acknowledgement of which caused the delay.

First, consider the expression for $T_3(0)$ and then for $T_3(w)$ for w = {1, 2, ..., (M-2)}. As shown in Figure 12, the acknowledgement for frame I0 is disturbed, and hence, transmission of frame I3 is delayed by time $T_3 = t_l$. T_3 is counted in the virtual transmission time of frame I0. If one of the following (M-2) I-frames is disturbed then T_3 is a part of time T_1 of the disturbed frame. Since the disturbed I-frame and all the following frames are be retransmitted, T_3 does not cause an effective delay. Figure 13 shows such a case. where it can be seen easily.

T_3 is accounted for if all the I-frames following the one under consideration are received successfully. The probability of this event is $(1 - P_B)^{M-2}$ and the sojourn time is t_l .

The situation where all the acknowledgement frames following the one under consideration are destroyed must now be considered. In such case, the time-out recovery is performed as shown in Figure 14.

The time T by which transmission of frame I3 was delayed is

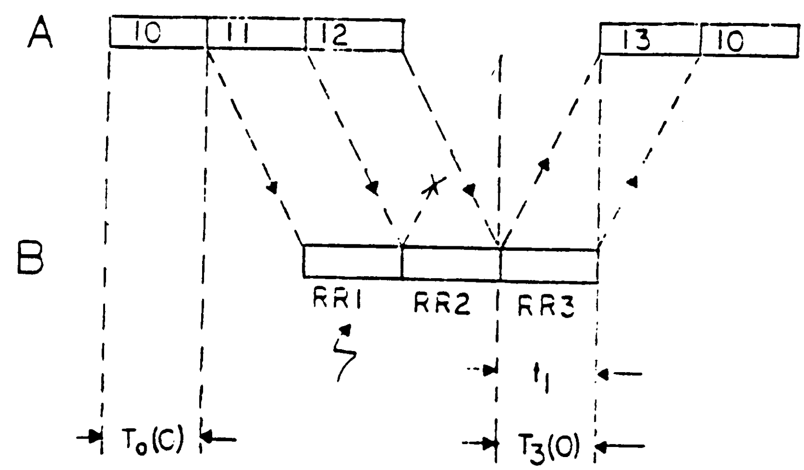


Figure 12. Time period T3

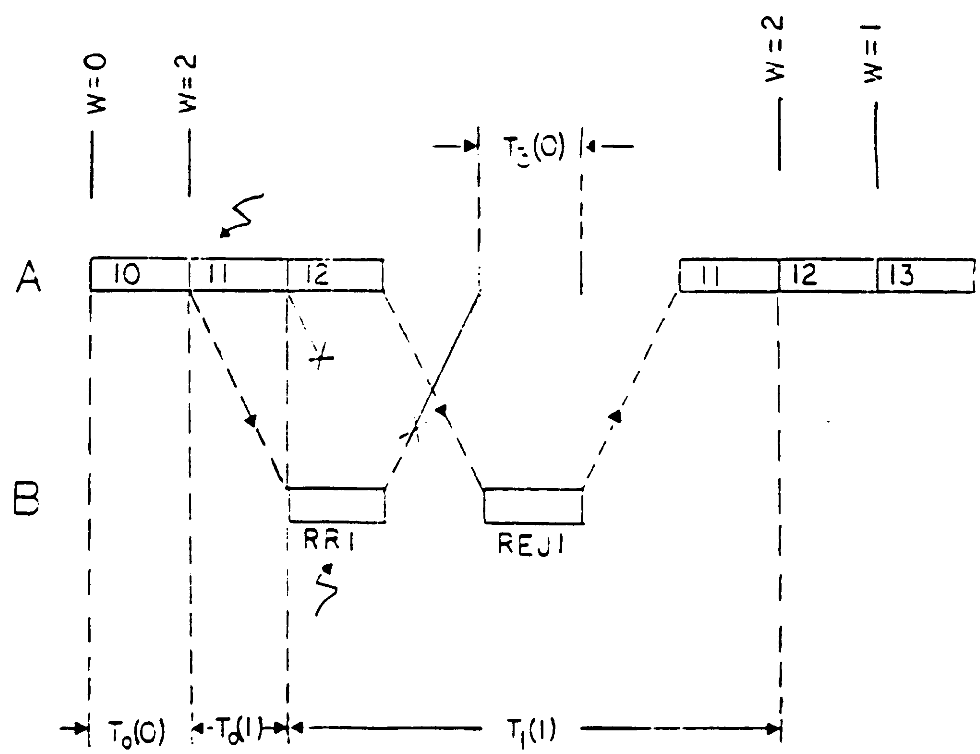


Figure 13. Additional considerations for T3

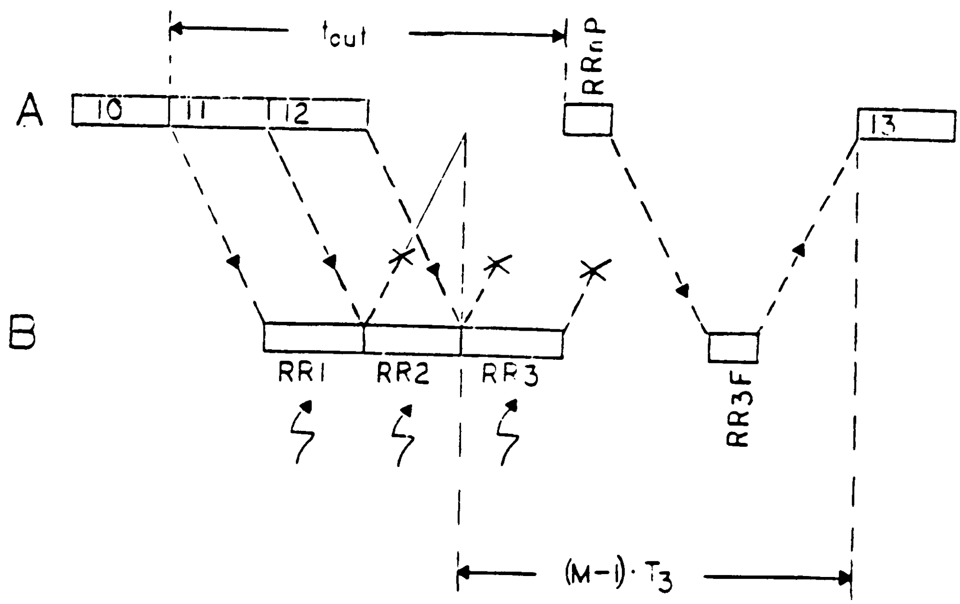


Figure 14. All acknowledgements disturbed

$$T = t_{out} - (M-2)t_l - t_d + t_s + t_{ack} \qquad [4.23]$$

But in time T, (M-1) frames were successfully transmitted so it is distributed equally among (M-1) frames.

The probability P of this event is

$$P = (1-P_B)^{M-2} P_B^{M-2} \qquad [4.24]$$

Thus the expected value $E(T_3(0))$ is given by

$$\begin{aligned} E(T_3(0)) &= (1-P_B)^{M-2} t_l \\ &\quad + (1-P_B)^{M-2} P_B^{M-2} (t_{out} - (M-2)t_l - t_d + t_s + t_{ack})/(M-1) \end{aligned} \qquad [4.25]$$

Consider now the situation for $T_3(w)$. As discussed in the section for $T_1(w)$, for I-frames with w = {1, 2, ..., (M-2)}, exactly (w-1) frames are transmitted before delay t_d following the frame under consideration. So, the acknowledgements for these frames are also spaced such that (w-1) acknowledgements arrive before the delay and (M-2)-(w-1) after the delay. Given an error in the acknowledgement for frame under consideration, if one of the first (w-1) acknowledgements goes through, then there is no delay in addotion to t_l. If all (w-1) acknowledgements are disturbed, and one of the remaining (M-2)-(w-1) acknowledgements is received, then t_d is counted in $T_3(w)$ as follows:

If x acknowledgements from (M-2)-(w-1) acknowledgements are disturbed, such that x ≤ (M-2)-(w-1)-1 and the next one is received, then refering to Figure 15 we see that additional time t_d is spent with probability P:

$$P = P_B^{(w-1)} \sum_{x=0}^{((M-2)-(w-1)-1)} P_B^x (1-P_B) \qquad [4.26]$$

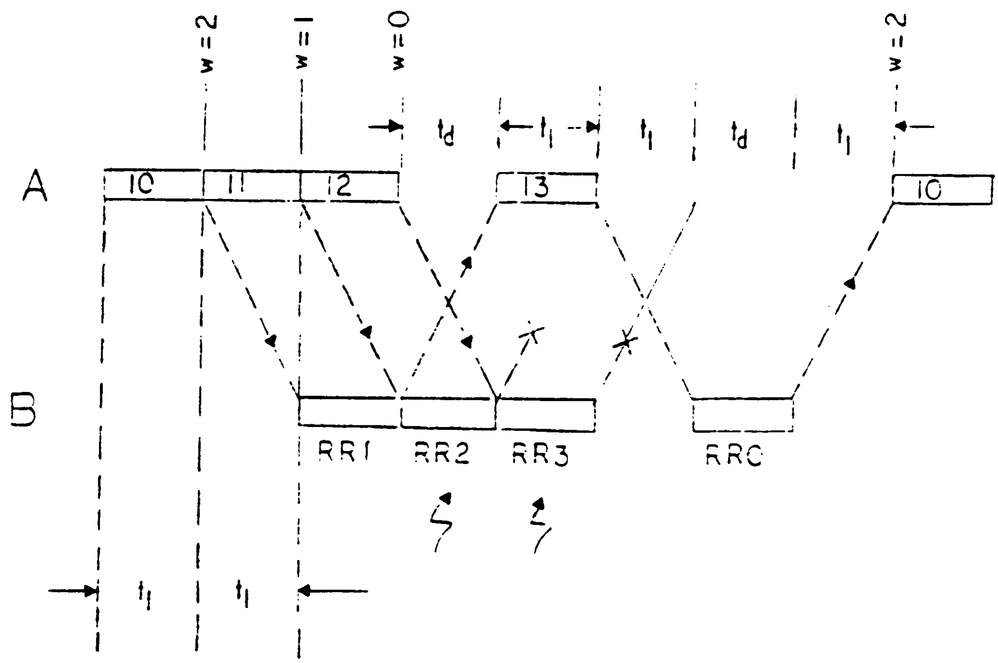


Figure 15. Time period T3(w)

But number of frames acknowledged by $(x+1)^{th}$ acknowledgement is $(w+x+1)$ so t_d is equally counted among $(w+x+1)$ frames.

Therefore the expected value $E(T_3(w))$ is

$$E(T_3(w)) = E(T_3(0)) + \left(P_B^{(w-1)} \sum_{x=0}^{((M-2)-(w-1)-1)} P_B^x (1-P_B) t_d\right) / (w+x+1) \qquad [4.27]$$

4.4 Case II)

The principle difference between this case and case I) is that the virtual transmission time does not depend on window size. Therefore the virtual transmission time t_v is given by

$$t_v = E(T_v) = t_l + P_B E(T_1) + P_B^2 \sum_{x=0}^{\infty} P_B^x (x+1)(1-P_B) T_2 + \sum_{x=0}^{\infty} P_B^x (1-P_B) P_B E(T_3) \qquad [4.28]$$

Here, the time periods T_1, T_2 and T_3 are found to be dependent on a parameter 'c', the number of I-frames which can be transmitted following a disturbed I-frame before recovery is performed. Clearly c is either (M-2), (the transmitter must stop because of (M-1) unacknowledged frames), or the number of I-frames that can be transmitted before time-out occurs, whichever is less. The number of I-frames 'n' that can be transmitted before time-out occurs is given by

$$n = \lfloor t_{out}/t_l \rfloor + 1 \qquad [4.29]$$

where $\lfloor a \rfloor$ is the largest integer not exceeding a.

Therefore,

$$c = \min\{M-2, \lfloor t_{out}/t_l \rfloor + 1\} \qquad [4.30]$$

Time period T_1 :

As in case I), T_1 is the time period from the end of first transmission to end of first retransmission. T_1 is determined depending on on whether or not the recovery is performed via time-out or REJ mechanism.

REJ RECOVERY

Recovery via REJ is performed when, not all c frames following a disturbed frame are disturbed and the acknowledgement (REJ) frame for the first I-frame is received undisturbed. Provided such an acknowledgement is received, T_1 depends on whether it arrives before or after the time-out occurs. The channel is busy transmitting I-frames before the time-out occurs and is idle after the time-out occurs when the acknowledgement is received. Both cases are considered in the derivation of expressions for T_1.

As shown in Figure 16,

$$T_1 = t_x = (x + 1)t_l + t_{ack} + t_l \qquad if\ (x + 1)t_l + t_{ack} > ct_l \tag{4.31}$$

Here, the number of frames disturbed is such that the transmitter is idle either due to expiration of time-out or (M-1) unacknowledged frames. Hence, retransmission of the frame would begin immediately upon receipt of REJ frame.

The probability P of x frames disturbed is

$$P = P_B^x(1 - P_B)(1 - P_B) \tag{4.32}$$

In the other case, a REJ frame arrives at the sender before time-out expires. The sender is busy transmitting an I-frame at this time. It finishes the transmission of current frame before retransmitting the disturbed frame. Figure 17 shows such situation.

Here packet I0 is shown disturbed and packet I5 is being transmitted when REJ0 arrives. The condition for this is $(x + 1)t_l + t_{ack} \le ct_l$

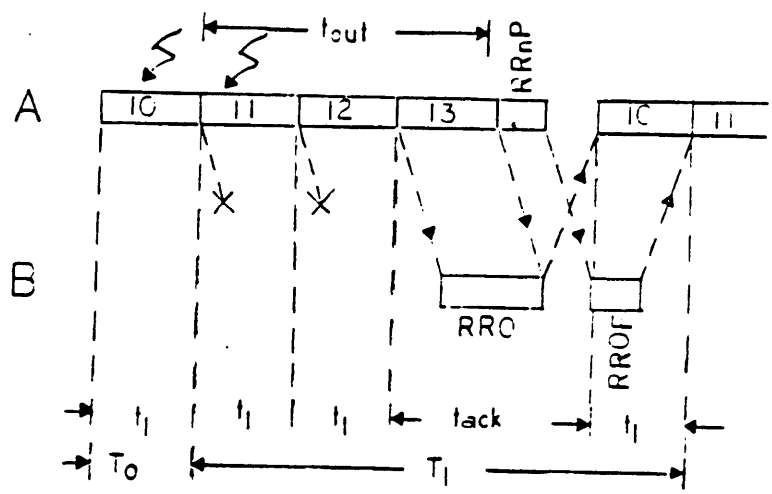


Figure 16. time-out small

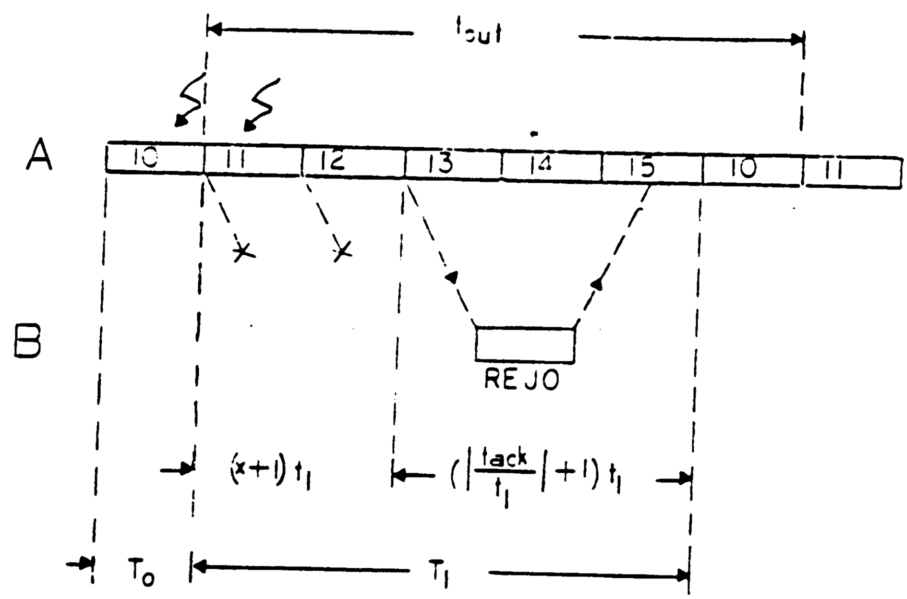


Figure 17. time-out large

T_1 in this case is given by

$$T_1 = t_x = (x+1)t_l + (\lfloor t_{ack}/t_l \rfloor + 1)t_l + t_l \qquad [4.33]$$

with probability P

$$P = P_B^x(1-P_B)(1-P_B) \qquad [4.34]$$

TIME-OUT RECOVERY

Time-out recovery is performed if all c I-frames following the one under consideration are disturbed, or the acknowledgement for the first I-frame received, is disturbed.

The probability P of this event is

$$P = P_B^c + \sum_{x=0}^{c-1} P_B^x(1-P_B)P_B \qquad [4.35]$$

Now for $t_{out} \le ct_l$,

$$T_1 = t_c = ct_l + t_s + t_{ack} + t_l \qquad [4.36]$$

where the time-out expires before all c I-frames are over, and recovery is started immediately after c^{th} I-frame.

and for $t_{out} > ct_l$,

$$T_1 = t_c = t_{out} + t_s + t_{ack} + t_l \qquad [4.37]$$

The sender waits until the time-out expires before starting recovery.

From equations above, it follows that

$$E(T_1) = \sum_{x=0}^{c-1} P_B^x(1 - P_B)(1 - P_B)t_x$$
$$+ (P_B^c + \sum_{x=0}^{c-1} P_B^x(1 - P_B)P_B)t_c \qquad [4.38]$$

Time period T_2 :

Since REJ recovery cannot take place on retransmissions, only time-out is performed. The expression for T_2 is same as time-out recovery expression for T_1. Thus,

$$T_2 = ct_l + t_s + t_{ack} + t_l \qquad if\ t_{out} \leq ct_l$$

$$= t_{out} + t_s + t_{ack} + t_l \qquad if\ t_{out} > ct_l$$

Time period T_3 :

T_3 is defined as the time for which the transmission of an I-frame is delayed due to a disturbed acknowledgement for a previously transmitted I-frame. We consider two cases for T_3, (a), $t_{out} > ct_l$ and (b), $t_{out} \leq ct_l$

CASE (a) $t_{out} > ct_l$

As shown in Figure 18, as long as one acknowledgement is received before all c I-frames are transmitted, then there is no delay in further transmissions. This occurs when I-frames transmitted before time $(ct_l - t_{ack})$ are received successfully, and at least one acknowledgement returns. The number of I-frames transmitted prior to time $(ct_l - T_{ack})$ is given as,

$$k = \lfloor((ct_l - t_{ack})/t_l)\rfloor \qquad [4.40]$$

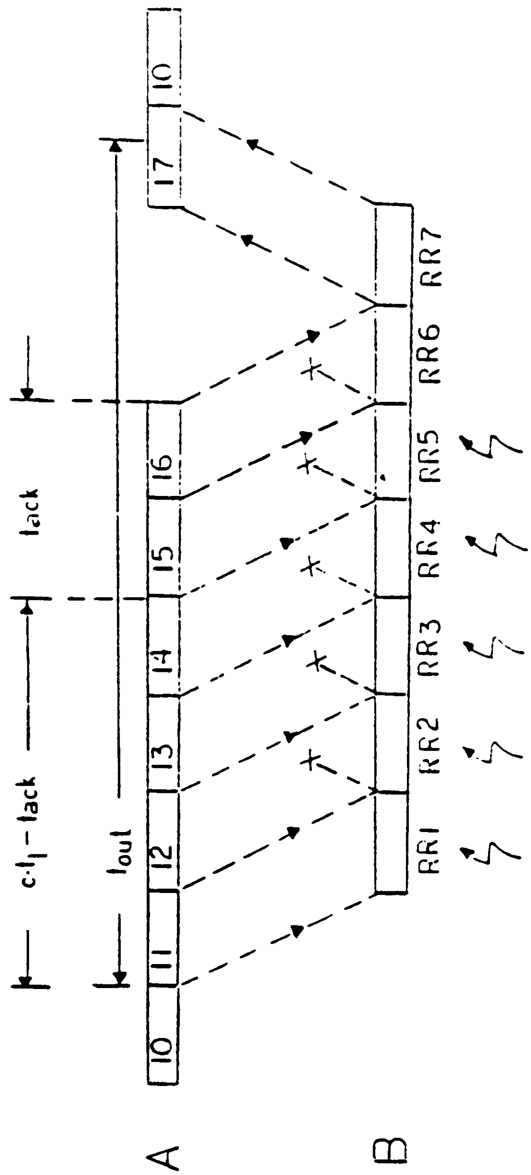


Figure 19. Time period T3

Thus with probability P_B^k, the system is delayed when some of the remaining acknowledgements are disturbed. More precisely, if x I-frames after k are disturbed such that k + x < c, then the time spent T in waiting for an acknowledgement is

$$T = (k + x + 1)t_I + t_{ack} - ct_I \qquad [4.41]$$

with probability P

$$P = P_B^{k+x}(1 - P_B)^c(1 - P_B) \qquad [4.42]$$

The number of I-frames that are successfully transmitted in this time is (k + x + 1), so this time is equally distributed in the t_v of each of these I-frames.
If all c I-frames are disturbed, then sojourn time T is,

$$T = (T_{out} + t_s + t_{ack} - ct_I) \qquad [4.43]$$

which is equally distributed in the t_v of c I-frames with probability $P_B^c(1 - P_B)^c$.
Therefore

$$E(T_3) = P_B^k \sum_{x=0}^{c-k-1} P_B^x(1 - P_B)^c(1 - P_B)t_3 + P_B^c(1 - P_B)^c(t_{out} + t_s + t_{ack} - ct_I)/c \qquad [4.44]$$

where

$$t_3 = ((k + x + 1)t_I + t_{ack} - ct_I)/(k + x + 1) \qquad [4.45]$$

CASE (b) $t_{out} \leq ct_I$

Analysis for T_3 is same as case (a), except if all c acknowledgements are disturbed. Since $(c - 1)t_I < t_{out} \leq ct_I$, recovery begins immediately after the c^{th} I-frame has completed transmission. The sojourn time is $t_s + t_{ack}$, which is equally divided among c I-frames.

Therefore,

$$E(T_3) = P_B^k \sum_{x=0}^{c-k-1} P_B^x (1 - P_B)^c (1 - P_B) t_3 + P_B^c (1 - P_B)^c (t_s + t_{ack})/c \qquad [4.46]$$

where

$$t_3 = ((k + x + 1)t_I + t_{ack} - ct_I)/(k + x + 1) \qquad [4.47]$$

This concludes the derivations of expressions for maximum throughput of information bits. In the next chapter, these expressions are plotted for various values of the relevant parameters.

5.0 Results and Conclusions

5.1 Results

In this chapter resulting expressions for throughput, derived in previous chapter are presented. The results of work by Bux et al. [8] are also plotted so as to enable comparison between their approach and the approach presented here. Each graph has two curves; the solid line represents the results of this approach and the dotted line represents results obtained by Bux et al. The results are plotted for various values of relevant parameters. This discussion is divided in to two parts: first, terrestrial links (one way propagation delay 50 ms) and second, satellite links (one way propagation delay 350 ms). Considered are, different error rates, window sizes and link capacities. The resulting plots are shown in Figure 19 through Figure 27. The Y-axis in these plots shows the relative throughput which is defined as (throughput/transmission rate).

First, consider the terrestrial links. The relative throughput of 48 kbps links is low for short packet sizes. There are two reasons for this: first, the data size is small and relative overhead of control bits is large. Second, for 48 kbps links, a window size of 8 is a setback since the I-frame transmission time is relatively low compared to round-trip propagation delay and the window is exhausted too early. This mode of operation wastes a substantial amount of bandwidth. However, as packet length increases, throughput increases. For very large packet lengths the performance degrades once again since the block error probability increases, and considerable time is lost in error recovery. For a transmission rate of 4800 bps the performance for small packet lengths is improved when compared with 48 kbps links since the window size proves adequate. The throughput increases with the packet length up to a certain maximum value and then drops again. For the same value of transmission rate, performance degrades as the error rate increases as shown in all plots. The discontinuities peculiar to 48 kbps links have the same explanation as offered by Bux et al [8]. The packet lengths for which t_{ack} is an integral multiple of t_I and is less than $(M-2)t_I$, show these discontinuities. In such case, acknowledgements are received at the sender,following the end of transmission of an I-frame; there is little delay in transmission of further frames. Thus, in the error recovery phase, upon receipt of an REJ frame, retransmission begins after a negligible delay. However, if REJ is received just after transmission of an I-frame begins, then retransmission is delayed by almost one packet transmission time, giving degraded throughput. This situation occurs at packet lengths for which t_{ack} is not an integral multiple of t_I.

Now consider satellite links. The one way propagation delay is taken to be 350 msec, and again results for relative throughput are plotted for various error rates and transmission rates. The explanation for these curves is similar to that of terrestrial links. The maximum throughput however is relatively less when compared with terrestrial links. The reason for this being that a longer time is spent in error recovery due to longer propagation delay.

These results are compared with results of Bux et al. [8] and the following observations are made.

For low values of error rate, the curves do not differ significantly. For higher error rates, the curves once again do not differ much at short packet lengths. However for longer packet lengths and higher error rates the difference is substantial. The reason for this is that the time component T_3 of the virtual transmission time, which depends on the errors in acks, is directly proportional to the block error rate and the transmission time of single I-frame. Both of these quantities increase as the frame length increases, and thus yield lower throughput. The effect of errors in acks on satellite links is exaggerated due to longer recovery times, thus causing degraded throughput at all packet lengths.

The effect of varying window size on throughput is also considered. For satellite links, curves are plotted for window sizes of 16 and 24. The throughput is relatively better than that of window size 8 and also, the difference in throughput is considerably reduced.

For all practical applications, where typical packet length varies from 1000 to 3000 bits/packet, the protocol does not operate efficiently on satellite links. For higher values of M, however, the the performance is relatively improved. Thus, for satellite links the overall throughput is substantially less in our case and hence proves to be more accurate.

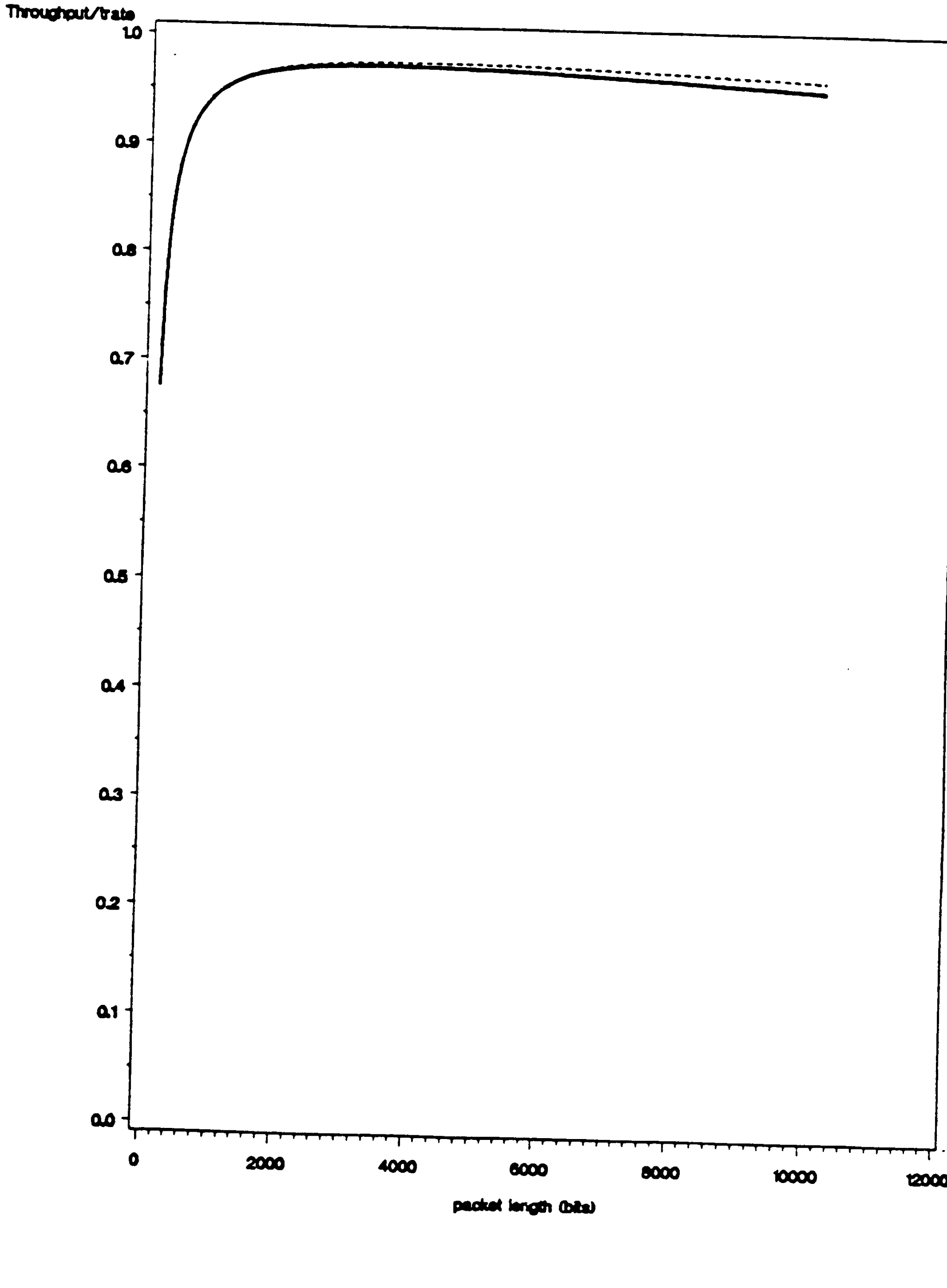


Figure 19. Pb=1.0e-6, 4800 bps, 50 msec, M=8

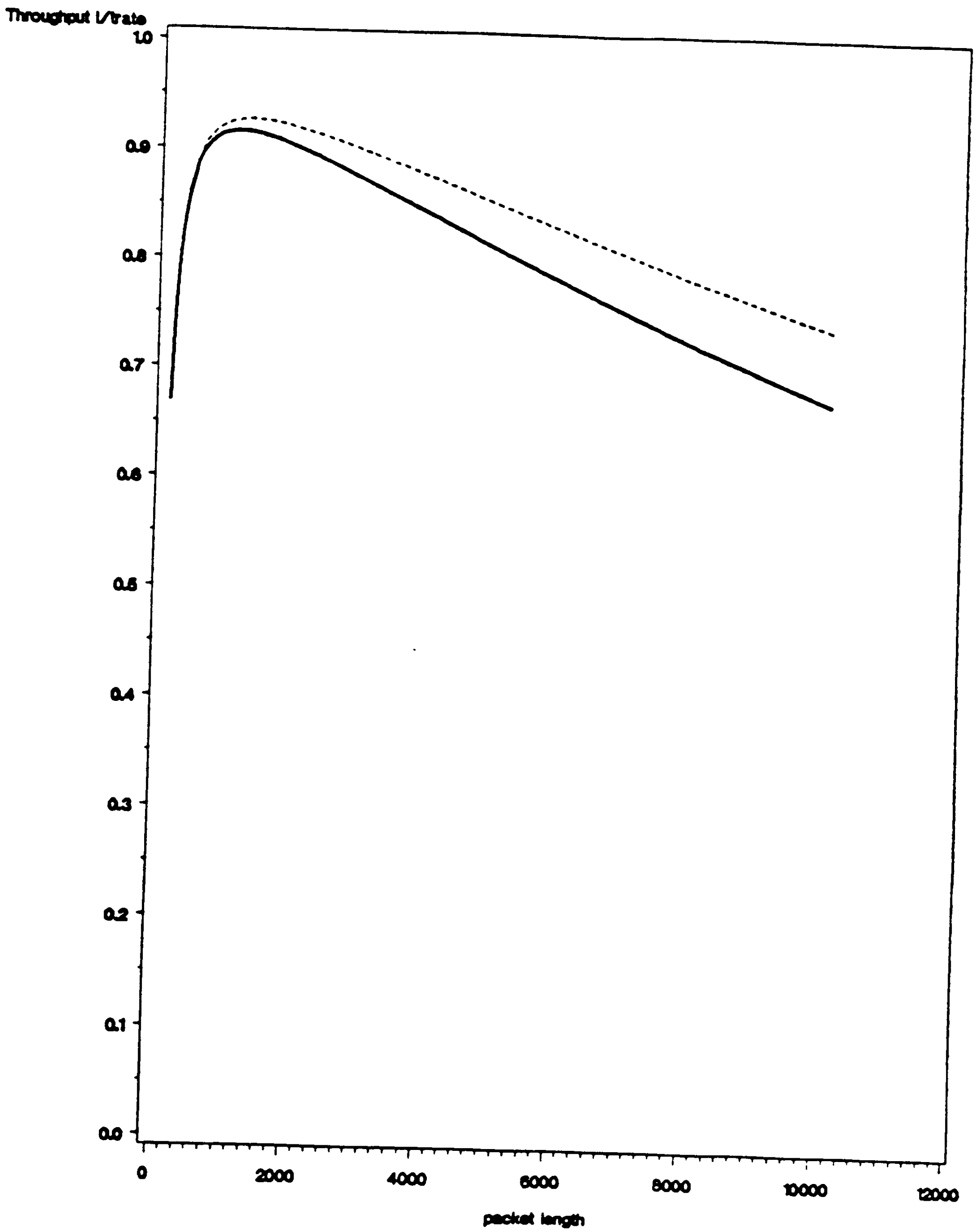


Figure 20. Pb=1.0e-5, 4800 bps, 50 msec, M=8

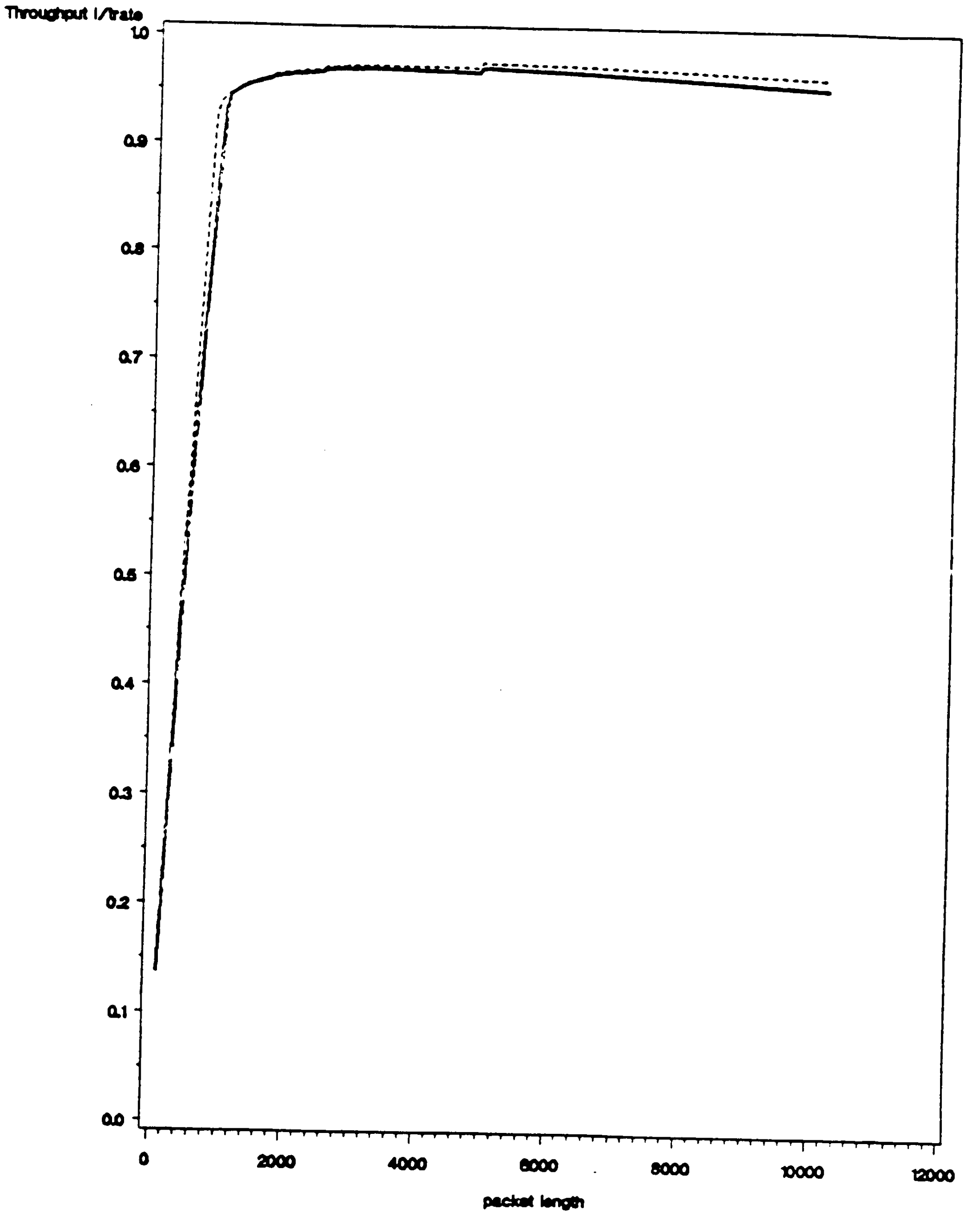


Figure 21. Pb=1.0e-6, 48 kbps, 50 msec, M=8

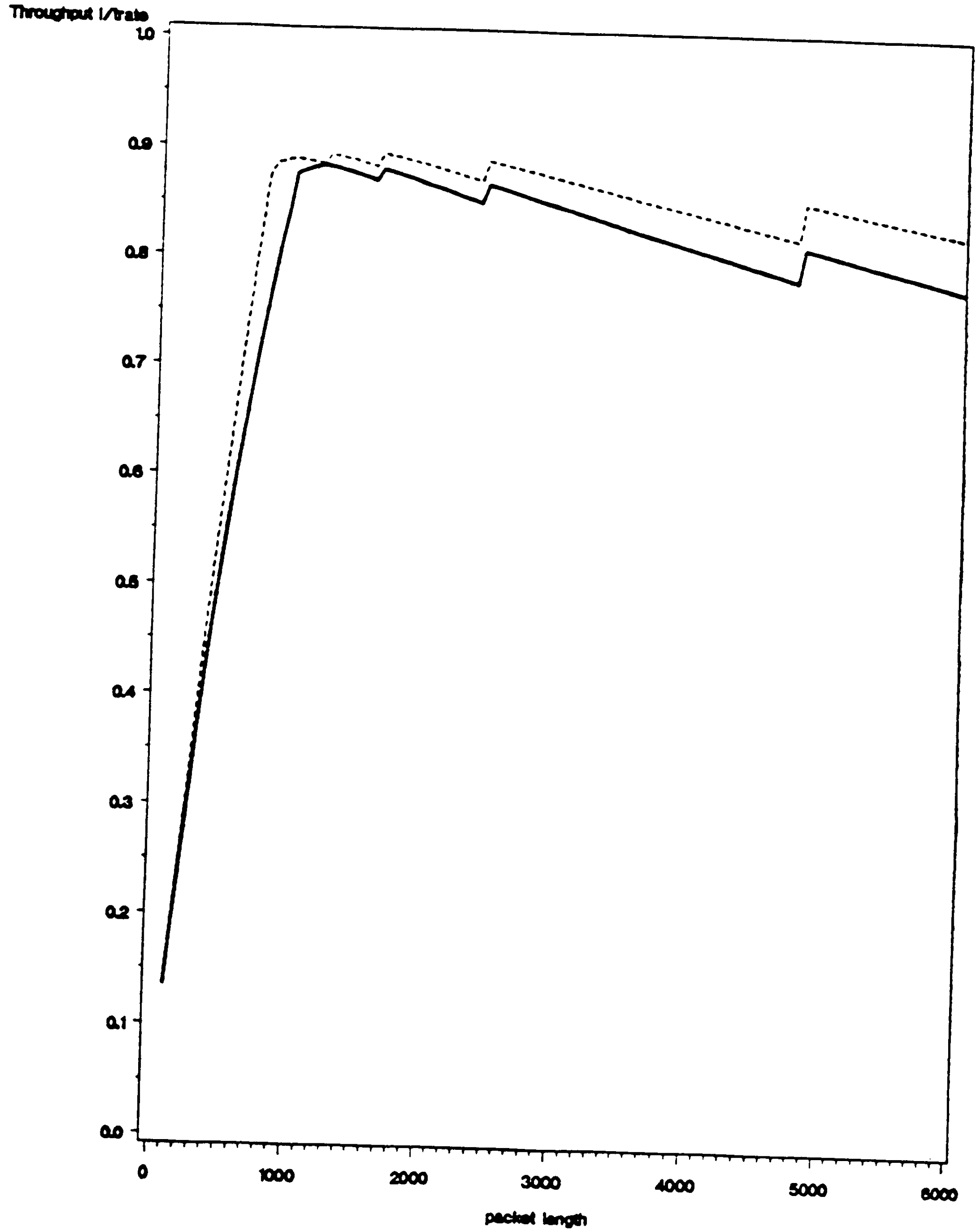


Figure 22. Pb=1.0e-5, 48 kbps, 50 msec, M=8

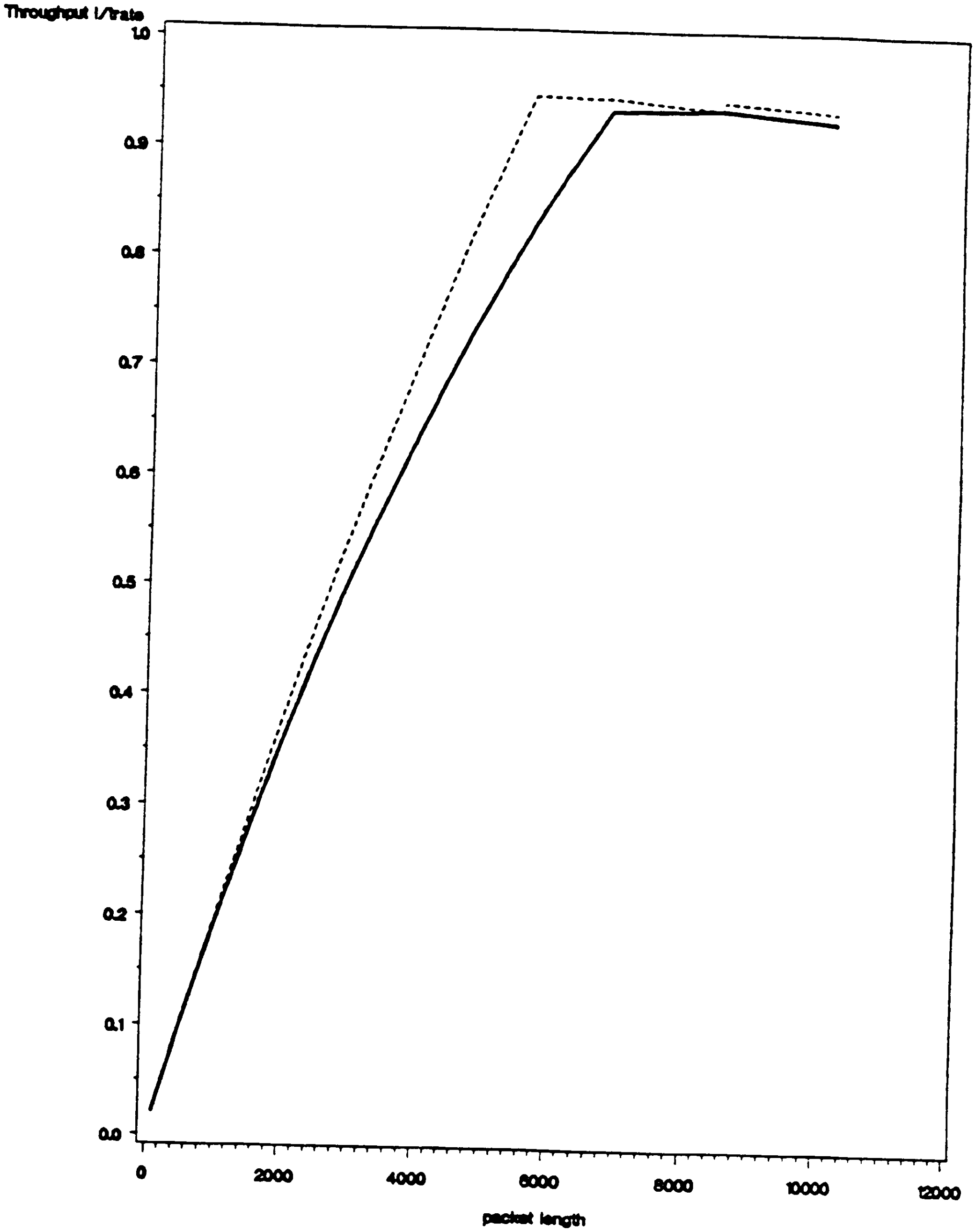


Figure 23. Pb=1.0e-6, 48 kbps, 350 msec, M=8

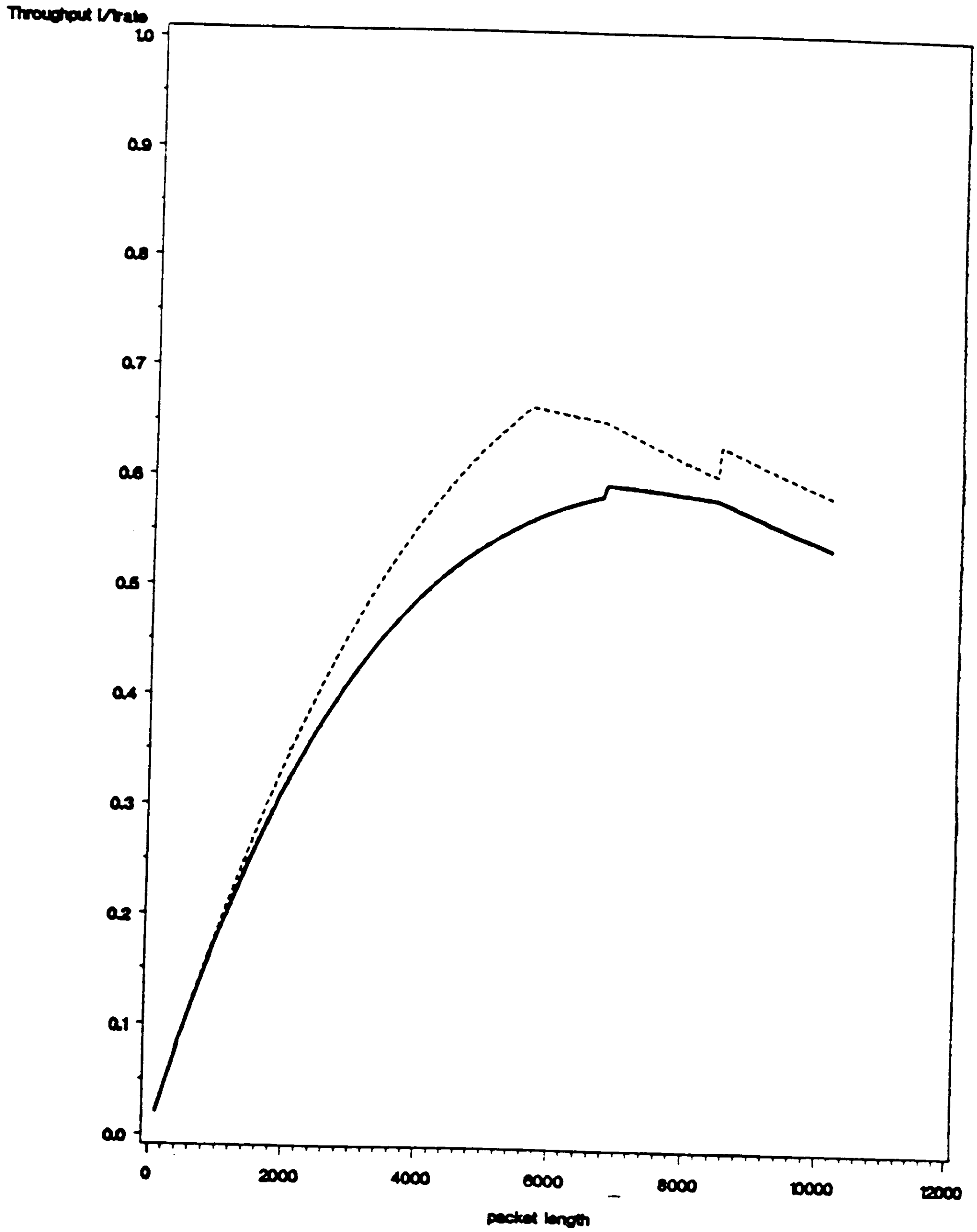


Figure 24. Pb=1.0e-5, 48 kbps, 350 msec, M=8

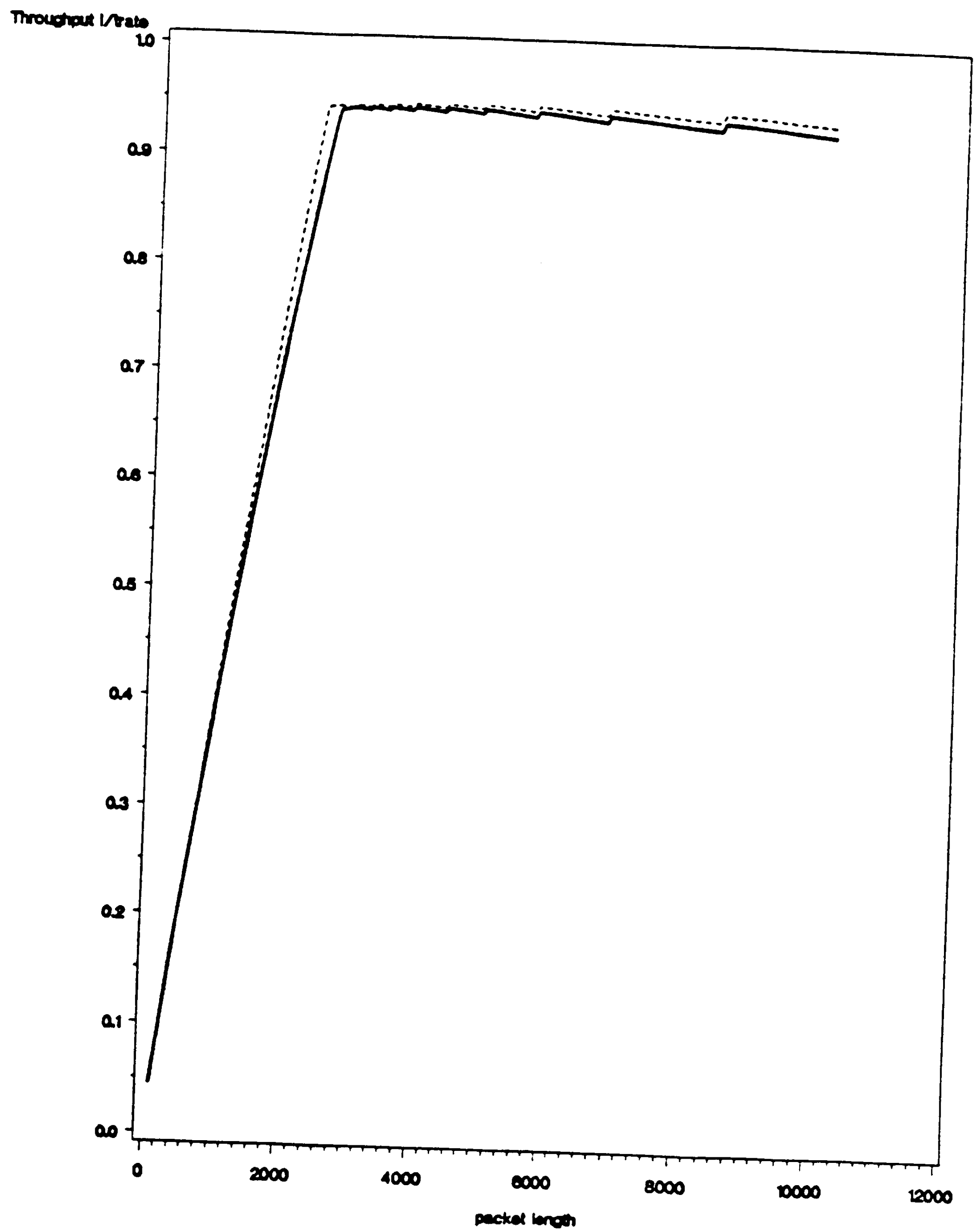


Figure 25. Pb=1.0e-6, 48 kbps, 350 msec, M=16

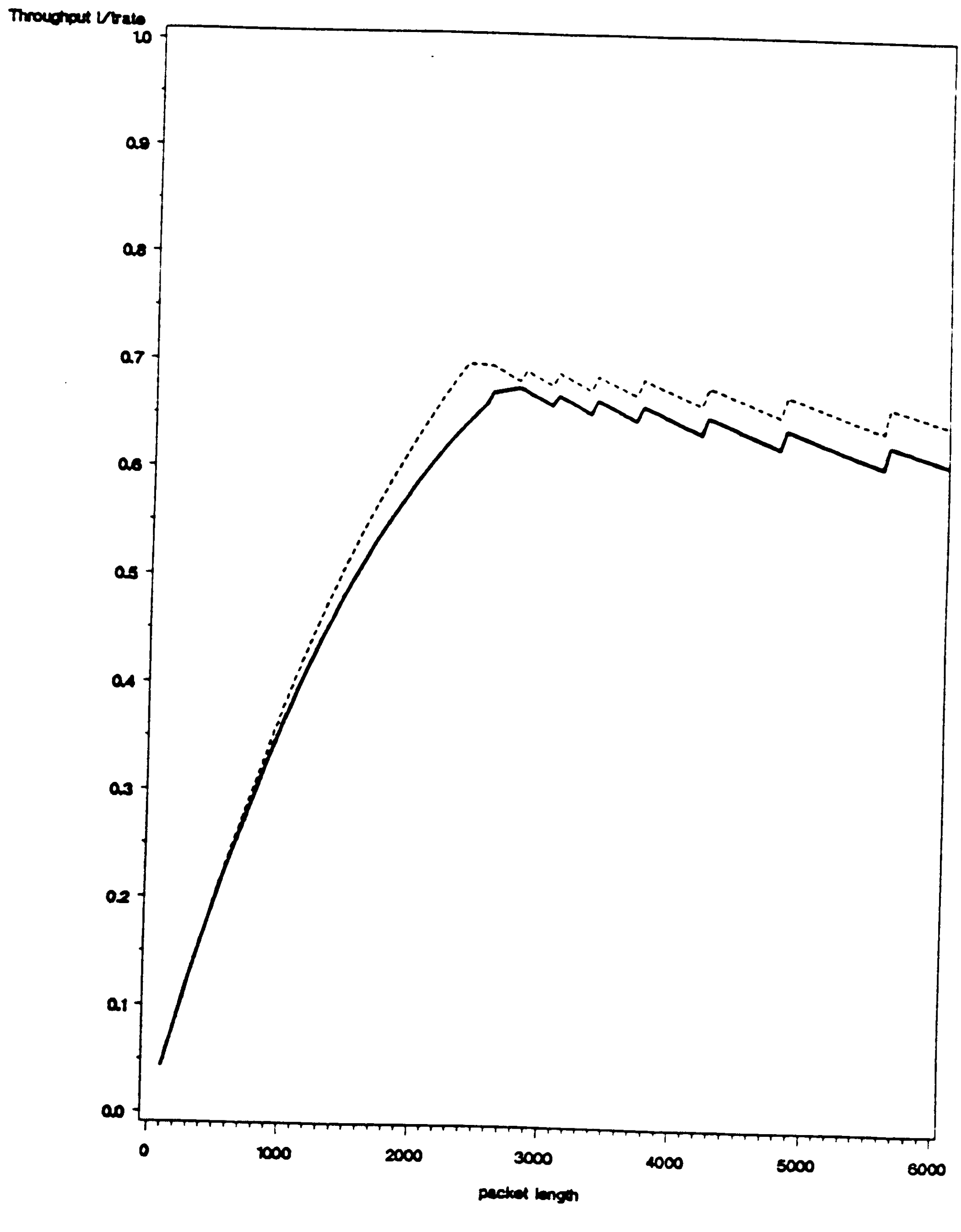


Figure 26. Pb=1.0e-5, 48 kbps, 350 msec, M=16

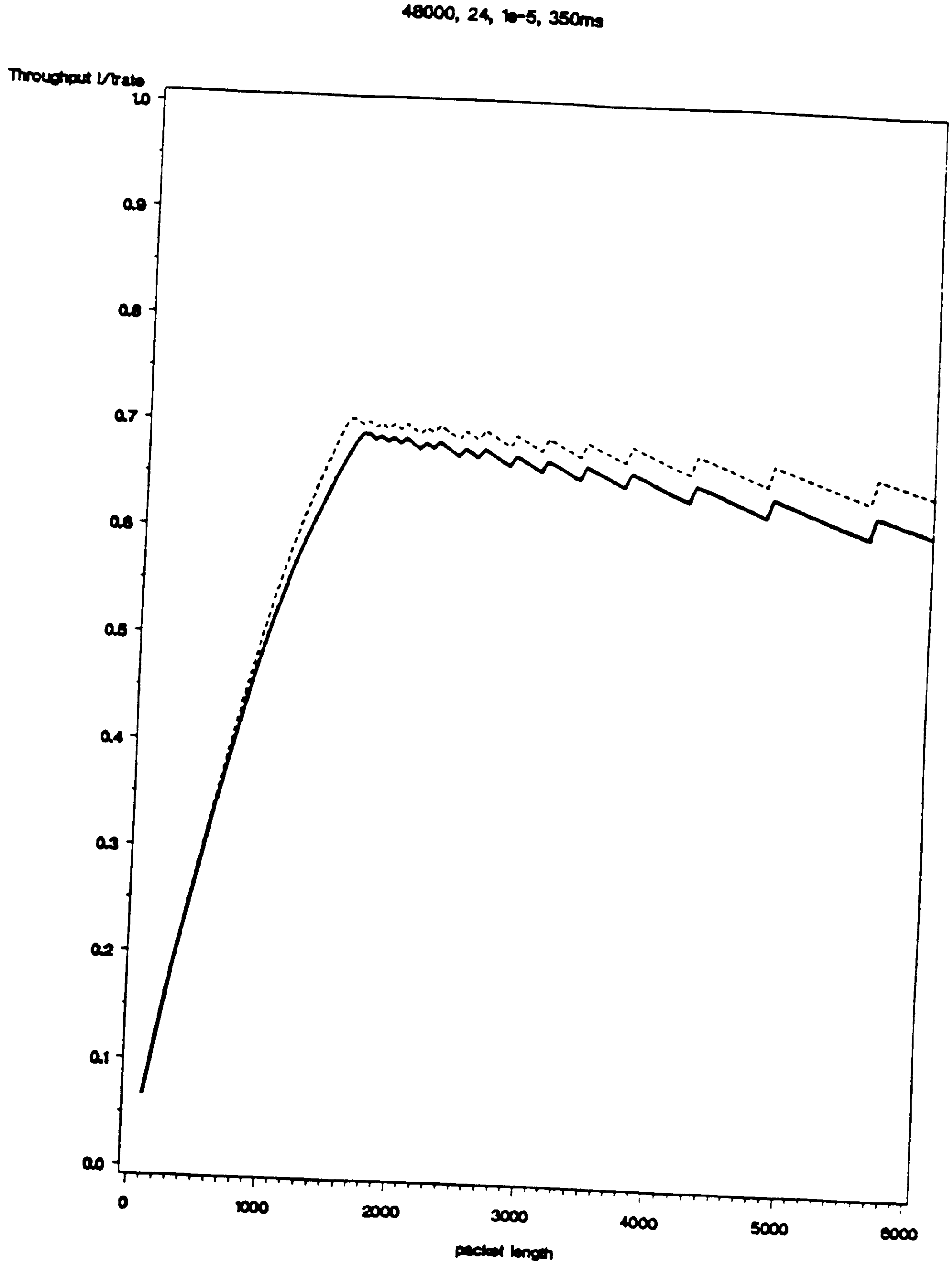


Figure 27. Pb=1.0e-5, 48 kbps, 350 msec, M=24

5.2 Conclusions

The principle objective of this work is to derive performance measures of the HDLC protocol operating in ABM mode on error prone links.

In conclusion, it is possible to analyze the HDLC protocol (ABM mode) using the redefined "virtual transmission time".

This analysis provides a better insight in to the operation of the protocol on error prone links. The errors in acknowledgements can have a substantial impact on the throughput for both terrestrial and satellite links. The protocol often appears inefficient on satellite links unless operated with a window size greater than 128. For terrestrial links, the throughput at short packet lengths appears insensitive.

Consider the ramifications of the assumptions made in this analysis. Although two key assumptions were made in order to simplify an otherwise tedious mathematical analysis, the results give a good insight in to the protocol opration.

One of the assumptions made in this analysis is that station B sends an information frame with piggybacked acknowledgement only upon receipt of an I-frame from station A. This means station B is idle whenever a frame from station A is received. This is not the situation in practice. Station B can transmit frames independently of station A. Hence, there will be instances where a frame from station A reaches station B when B is busy transmitting a frame. In such cases, a response will be delayed until B finishes transmitting that frame. The maximum delay caused by this action will be t_f (transmission time of single I-frame). Thus, t_{ack} will be greater than $2t_p + t_f$ causing degraded throughput. However, the difference in throughput will not be significant at short packet lengths (t_f is small) and the assumption stated above should yield acceptable results.

Second assumption key to the analysis is that both the stations are saturated in data. If this assumption is relaxed and we consider unsaturated data traffic, a probability distribution for inter arrival time of data packets must be specified in the analysis. The throughput in such case is defined as "transfer time" of a packet which is the time from arrival to departure of a packet from the input buffer. This requires a different approach to solve the problem. The assumption stated above simplifies the queueing model (the server is always busy) and allows the upper bound on throughput to be calculated.

Further research on HDLC (ABM mode of operation) can be done by removing the assumptions stated above. The transfer time results can provide a better insight in to the operation of the protocol on error prone links. The ABM mode of operation does not provide use of "selective reject" for error recovery, however, it can be analyzed assuming that "selective reject" is used for error recovery. A comparison between "go-back-n" and "selective reject" can show how efficiently the "go-back-n" performs on satellite links. This will enable us to consider the necessity of "selective reject".

Bibliography

1. A. Tanenbaum, *Computer Networks*, Prentice Hall, 1981.

2. M. Schwartz, *Telecommunication Networks: Protocols, Modeling and Analysis*, Addison Wesley, 1987.

3. D. Towsley and J. K. Wolf, *"On the Statistical Analysis of Queue lengths and waiting times for statistical multiplexers with ARQ retransmission schemes,"* IEEE Transactions on Communications, vol. COM-27, no. 4, pp. 693-702, April 1979.

4. A. G. Konheim, *"A Queueing Analysis of two ARQ Protocols,"* IEEE Transactions on Communications, vol. COM-28, no. 7, pp. 1004-1014, July 1980.

5. K. Sabnani and M. Schwartz, *"Performance Evaluation of Multidestination (Broadcast) Protocols for Satellite Transmission,"* National Telecommunications Conference, New Orleans, Nov. 1981.

6. L. W. Yu and J. Majithia, *"An Analysis of One Direction of Window mechanism,"* IEEE Transactions on Communications, vol. COM-27, no. 5, pp. 778-788, May 1979.

7. D. Davis, D. Barber, W. Price and C. Solomonides, *Computer Networks and Their Protocols*, John Wiley & Sons, 1979.

8. W. Bux, K. Kummerle and H. L. Truong, *"Balanced HDLC Procedures: A Performance Analysis,"* IEEE Transactions on Communications, vol. COM-28, no. 11, pp. 1889-1898, Nov. 1980.

9. Wang J., *"Delay and Throughput Analysis for Communication Networks with Balanced HDLC Procedures,"* IEEE Transactions on Computers, vol. COM-31, no. 8, pp. 739-746, 1982.

10. Gopal I., *"Point to Multipoint Communication over Broadcast Links,"* IEEE Transactions on Communications, vol. COM-32, no. 9, pp. 1034-1044, 1984.

11. M. Schwartz, *Computer Communication Network Design & Analysis*, Prentice-Hall, 1977.

12. D. E. Carlson, *"Bit Oriented Data Link Control Procedures,"* IEEE Transactions on Communications, vol. COM-28, no. 4, pp. 455-467, April 1980.

13. Shu Lin and Daniel J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*, Prentice-Hall, 1983.

The vita has been removed from the scanned document