

PH.D. THESIS

Error Control for Multicasting in Satellite and Hybrid Communication Networks

by Daniel E. Friedman

Advisor: Anthony Ephremides

**CSHCN PhD 2001-2
(ISR PhD 2001-4)**



The Center for Satellite and Hybrid Communication Networks is a NASA-sponsored Commercial Space Center also supported by the Department of Defense (DOD), industry, the State of Maryland, the University of Maryland and the Institute for Systems Research. This document is a technical report in the CSHCN series originating at the University of Maryland.

Web site <http://www.isr.umd.edu/CSHCN/>

Abstract

Title of Dissertation: Error Control for Multicasting in Satellite and Hybrid
Communication Networks

Daniel E. Friedman, Doctor of Philosophy, 2000

Dissertation directed by: Professor Anthony Ephremides
Department of Electrical Engineering

A problem inherent in ARQ multicasting over a broadcast channel is that a retransmission typically benefits only a minority of destinations while all others wait unproductively. This results in poor throughput to each receiving station in the network, with the throughput diminishing as the number of receivers grows.

If point-to-point links between the transmitter and each receiver were also available, then conceivably retransmissions could be sent over such secondary links. This would reduce the frequency of retransmissions interrupting the flow of new packets on the broadcast link. That is, a *hybrid* satellite-terrestrial network architecture would allow greater throughput for multicasting than a pure-satellite network.

This work examines ARQ multicasting in such a network, and confirms by analysis and simulation that, within limits, such a throughput advantage can be realized. A detailed discussion of implementation aspects for point-to-point and point-to-multipoint ARQ protocols in both pure-satellite and hybrid networks is presented as

well. This work also considers partitioning a fixed amount of bandwidth to maximize throughput, possibly subject to a cost constraint, and the effect of a “poor listener” upon performance in both pure-satellite and hybrid networks.

Error Control for Multicasting in Satellite and Hybrid Communication Networks

by

Daniel E. Friedman

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2000

Advisory Committee:

Professor Anthony Ephremides, Chairman/Advisor
Professor John Baras
Professor Evaggelos Geraniotis
Professor Leana Golubchik
Doctor A. Roger Hammons, Jr.

© Copyright by
Daniel E. Friedman
2000

Acknowledgements

I wish to express my thanks to some of the many individuals who made this work possible. Foremost among these is my advisor, Professor Anthony Ephremides, who has provided me excellent opportunities for learning and patiently given me invaluable guidance. I was particularly privileged to work with NASA's Advanced Communications Technology Satellite (ACTS) for two years as part of an experiments program supervised by Dr. Ephremides. That work led to my M.S. degree and provided valuable experience for the research presented in this dissertation.

I am also grateful to Professor John Baras (Electrical and Computer Engineering Department/Institute for Systems Research), Professor Evaggelos Geraniotis (also ECE/ISR), Professor Leana Golubchik (Computer Science Department/University of Maryland Institute for Advanced Computer Studies), and Dr. A. Roger Hammons, Jr. (formerly at Hughes Network Systems, Germantown, MD; now at Corvis Corporation, Columbia

MD), for kindly consenting to join the defense committee and review this dissertation. Special additional thanks to Dr. Hammons for eagerly joining the committee only one month before the defense after another member became unavailable.

I have been fortunate to work professionally for talented and kind friends during the latter portion of my graduate career. While at WorldSpace Corporation (Washington, DC), I was privileged to work for Dr. S. Joseph Campanella, a pioneer and renowned authority in the field of satellite communication, and Mr. Timothy Kirkwood (whom I met at the Center for Satellite and Hybrid Communication Networks at the University of Maryland, and who is now at Hughes Network Systems), a keen and treasured friend. More recently, I have renewed a wonderful relationship established during an internship four years ago with Dr. Xiang Chen at Hughes Network Systems. At this time, I'm working on an exciting project under the sage and avuncular direction of Mr. Charles Box and Dr. John Ashworth. I am grateful for what I have learned, and continue to learn, from these individuals, and also appreciate their granting me time to complete this work. Special additional thanks to Tim and John for reviewing this dissertation and offering many helpful comments.

Shortly after I began my graduate studies, I met Judah Milgram (Ph.D. 1997, Aeronautical Engineering, University of Maryland, College Park; now at Naval Surface Warfare Center, Carderock Division, West Bethesda, MD). He quickly became a close friend. I wish to thank Judah for introducing me to Linux, which became an indispensable convenience for this work, and for his broad wisdom and fine wit, in matters pedestrian

to arcane.

Thanks also to David Lindsay for helpful discussions regarding section 2.2.1.

This work was supported in part by the Graduate School of the University of Maryland at College Park through a graduate fellowship, by the Center for Satellite and Hybrid Communication Networks and the Institute for Systems Research through a research assistantship, and by Hughes Network Systems.

Finally, I wish to thank my family and friends—especially dear IYS—for all their support, patience, and love.

Table of Contents

<u>Section</u>	<u>Page</u>
List of Tables	viii
List of Figures	ix
1 Introduction	1
1.1 Motivation	1
1.2 Historical Perspective	4
1.2.1 Stop-and-Wait Multicast ARQ	4
1.2.2 Go-back- N Multicast ARQ	4
1.2.3 Improved GBN Multicast ARQ	6
1.2.4 Selective Repeat Multicast ARQ	7
1.2.5 Hybrid ARQ for Multicasting	7
1.2.6 Other Multicast ARQ Techniques	9
1.2.7 Placement of this Dissertation	10
1.3 Overview of the Dissertation	11
2 Throughput Analysis	12
2.1 Point-to-Point Communication	13

2.1.1	Pure-Satellite Network	13
2.1.2	Hybrid Network	15
2.2	Point-to-Multipoint Communication	19
2.2.1	Pure-Satellite Network	20
2.2.2	Hybrid Network	26
2.3	Numerical Examples	26
3	Implementation Considerations	32
3.1	Chapter Overview	32
3.2	Protocol Description	33
3.3	Parameter Settings	39
3.3.1	Information Frame Length	40
3.3.2	Timer Periods	50
3.3.3	Window Size	60
3.3.4	Maximum Acknowledgement Length	66
3.3.5	Summary for Setting ARQ Protocol Parameters	71
4	Results and Discussion	76
4.1	Numerical Example	76
4.1.1	Assumptions for Numerical Example	76
4.1.2	Single-Receiver Pure-Satellite Network	78
4.1.3	Multiple-Receiver Pure-Satellite Network	87
4.1.4	Single-Receiver Hybrid Network	90
4.1.5	Multiple-Receiver Hybrid Network	106
4.1.6	Comparison of Pure-Satellite and Hybrid Networks for Mul- ticast to Identical Receivers	113

4.2	Bandwidth Splitting	116
4.2.1	Problem Formulation	116
4.2.2	Relation of Splitting Factor to Throughput	117
4.2.3	Summary	123
4.2.4	Numerical Example	123
4.3	Multicasting to Non-Identical Receivers: The “Poor Listener” Problem	127
5	Conclusion	134
5.1	Dissertation Review	134
5.2	Issues for Future Study	135
A	Useful Identities	138
B	Glossary of Notation	141

List of Tables

<u>Number</u>	<u>Page</u>
4.1 Assumptions for Numerical Example	77

List of Figures

<u>Number</u>	<u>Page</u>
1.1 Multicasting in a hybrid network.	3
2.1 Information frame structure (as assumed for analysis).	14
2.2 Information flow in a unicast pure-satellite network.	14
2.3 Information flow in a unicast hybrid network.	16
2.4 Packet flow model for the hybrid network.	17
2.5 Throughput in point-to-multipoint networks.	28
2.6 Throughput efficiency in point-to-multipoint networks	30
3.1 Information frame structure (as implemented).	34
3.2 Acknowledgement composition.	36
3.3 Acknowledgement generation and loss at the receiver.	55
3.4 Parameter calculations for pure-satellite network, steps 1 to 3 (of 5).	72
3.5 Parameter calculations for pure-satellite network, steps 4 and 5 (of 5).	73
3.6 Parameter calculations for hybrid network, steps 1 and 2 (of 5).	74
3.7 Parameter calculations for hybrid network, steps 3 to 5 (of 5).	75
4.1 Optimal packet length for ARQ operation in the single-receiver pure-satellite network.	79

4.2	Satellite Channel FER, p_s , as calculated from optimal frame lengths.	81
4.3	Acknowledgement sequence numbers maximum, θ , for the pure-satellite network.	82
4.4	Throughput effects of θ in the pure-satellite network.	83
4.5	Values for ω in single-receiver, pure-satellite network.	85
4.6	Throughput effects of ω in the single-receiver, pure-satellite network.	86
4.7	Optimal packet length ℓ^* for ARQ multicasting in a pure-satellite network	88
4.8	Throughputs for multiple-receiver pure-satellite networks.	89
4.9	Optimal packet length for ARQ operation in the hybrid network.	91
4.10	Satellite channel FER, p_s , corresponding to optimal packet lengths.	92
4.11	Terrestrial channel FER, p_t , corresponding to optimal packet lengths.	93
4.12	Predicted throughput for ARQ operation in the hybrid network, using optimal packet lengths from Figure 4.9.	95
4.13	Acknowledgement sequence numbers maximum, θ , for the hybrid network.	96
4.14	Throughput effects of θ in the hybrid network.	97
4.15	Values of ω for the hybrid network.	99
4.16	Throughput effects of ω in the hybrid network	100
4.17	Optimal packet length, ℓ , for stable operation in the hybrid network.	102
4.18	Satellite channel FER, p_s , corresponding to the optimal packet lengths calculated to assure stable operation in the hybrid network	103
4.19	Terrestrial channel FER, p_t , corresponding to the optimal packet lengths calculated to assure stable operation in the hybrid network	104

4.20	Predicted throughput for ARQ operation in the hybrid network, using optimal packet lengths from Figure 4.17.	105
4.21	Acknowledgement sequence numbers maximum, θ , for the hybrid network, with ℓ calculated to assure stable operation.	107
4.22	Throughput effects of θ in the hybrid network with ℓ calculated to assure stable operation.	108
4.23	Values for ω in the hybrid network ($q_t = 10^{-5}$) with ℓ calculated to assure stable operation.	109
4.24	Throughput effects of ω in the hybrid network ($q_t = 10^{-5}$) with ℓ calculated to assure stable operation.	110
4.25	Throughputs for multiple-receiver hybrid networks.	112
4.26	Throughput comparison for the pure-satellite and hybrid networks. . .	114
4.27	Throughput efficiency comparison for the pure-satellite and hybrid networks.	115
4.28	Sketch showing variation of throughput, v , with splitting factor, ρ . . .	118
4.29	Results for example examining bandwidth splitting.	126
4.30	Throughput for “poor-listener” scenarios.	132

Error Control for Multicasting in Satellite and Hybrid
Communication Networks

Daniel E. Friedman

December 4, 2000

This comment page is not part of the dissertation.

Typeset by \LaTeX using the `dissertation` class by Pablo A. Straub, University of Maryland.

Chapter 1

Introduction

1.1 Motivation

Satellites are excellently suited for distributing information simultaneously to multiple locations. As in nearly all communication systems, some sort of error control scheme is required in satellite multicasting to assure satisfactory fidelity of the information provided to each destination.

Error control schemes may be broadly classified as forward error correction (FEC) or automatic-repeat-request (ARQ), and both can be applied for satellite communication. FEC has been used in satellite/space communication for decades, having grown from successful application by NASA for communication with interplanetary probes [1, 2]. However, satellite channel characteristics vary with time, and at any given time multiple receivers may perceive different channel qualities. Applying FEC for satellite multicast communication accordingly requires using an error-correcting code strong enough to protect data against worst-case channel impairments. Unfortunately the error correction capability provided by a powerful FEC code comes at the cost of sending many check symbols which constitute overhead in the communica-

tion. Further, this overhead penalty is exacted even at times of good channel quality, since FEC is not an adaptive error control technique. This is particularly troubling since good channel conditions will be experienced a majority of the time when using a well-designed satellite link [3].

ARQ protocols adapt to different channel qualities by retransmitting data only as needed. Also, an error-detecting code capable of detecting t or fewer errors in k information symbols requires fewer overhead symbols than would an FEC code designed to correct t errors in the same k symbols [1, 2]. Hence ARQ can provide high fidelity with less overhead than FEC during times of good channel quality, which tend to prevail as mentioned above. A drawback of ARQ not suffered by FEC is the need for a feedback channel, but this requirement is often an acceptable concession for achieving information transfer with excellent fidelity. Also, the presence of a feedback channel opens the possibility for so-called *reliable* communication, in which the transmitter can know positively that the receiver has indeed received information which the transmitter had sought to deliver, in correct order, without duplicates or errors.

A difficulty arises in applying ARQ in multicast settings. The typical problem in a multicast ARQ system is that retransmissions are sent over the multicast channel, but those retransmissions are typically required by only a few receivers and do not benefit the other receivers. The other receivers wait for a new information frame unproductively during such retransmissions. Accordingly the throughput for the system falls drastically as the number of receivers increases. Furthermore, if one receiving station is a “poorer listener” than other stations, namely it suffers a relatively high frame error rate, then the throughput to all stations is essentially limited by that poorer listener [4].

The throughput might be improved considerably if the retransmissions could some-

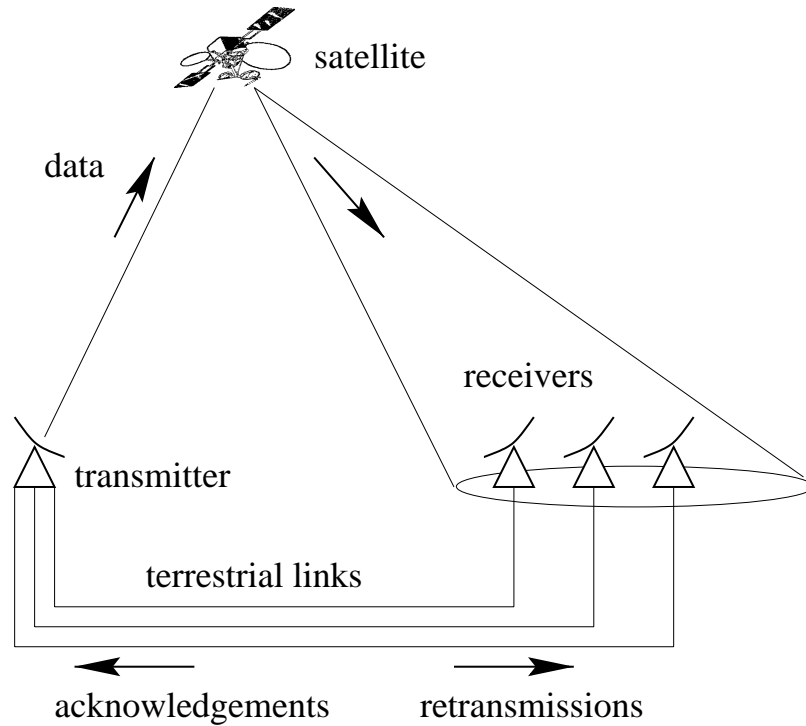


Figure 1.1: Multicasting in a hybrid network.

how be sent only to the receivers which need them. It is natural, then, to suggest supplementing a satellite multicast system with a set of point-to-point terrestrial links between the transmitter and each receiver, as depicted in Figure 1.1. In such a system, retransmissions could be sent terrestrially instead of via the multicast satellite link. This would allow retransmissions to be conducted without disrupting the flow of fresh information frames via the satellite link, and the throughput would possibly be thereby improved. Furthermore, if the ARQ acknowledgements were to be sent terrestrially as well, then the receiving stations would not require satellite transmission capability and the cost of such stations might be correspondingly reduced.

This dissertation examines multicast ARQ operation in such a *hybrid* (satellite and terrestrial) network configuration. A review of relevant literature is now presented to provide context for this work.

1.2 Historical Perspective

1.2.1 Stop-and-Wait Multicast ARQ

The seminal paper in multicast ARQ was written by Calo and Easton and appeared in 1981 [5]. This work presented a protocol, adapted from a standard stop-and-wait (SW) point-to-point ARQ protocol, for ARQ multicasting via satellite. In this protocol, the transmitter sends a block of N frames and then waits to collect acknowledgements. Each receiver's acknowledgement comprises a field identifying the receiver and an N -bit field to acknowledge positively and negatively the individual frames of the block just received. After waiting enough time to collect acknowledgements from all receivers, the transmitter sends a block of those frames not yet acknowledged by all receivers. The receivers again send acknowledgements and this process continues until all N frames have been acknowledged by all receivers.

1.2.2 Go-back- N Multicast ARQ

While Calo and Easton's scheme is a good start to addressing the need for ARQ multicasting, it inherits the inefficiency of stop-and-wait ARQ. Mase *et al.* proposed broadcast go-back- N (GBN) ARQ protocols for satellite communication in [6]. In particular, both an end-to-end protocol and a tandem protocol were discussed.

The end-to-end scheme is a modified "conventional multicast GBN ARQ protocol." Two features of such a "conventional" protocol are (1) every frame must be acknowledged, either positively or negatively, by all receivers; and (2) every frame transmitted must be positively acknowledged by all receivers in order to avoid a retransmission. Thus, if receiver A successfully receives a frame, but receiver B does not, and the frame is retransmitted, but this time receiver A does not successfully re-

ceive the retransmitted frame and so sends a negative acknowledgement, another retransmission is required. The authors attempt to avoid this inefficiency by modifying the “conventional” protocol: a special *retransmission indication block (RIB)* is introduced, to notify all receivers before a GBN retransmission commences. While not explicitly stated, it appears that perfect receipt of this RIB is assumed. (The authors do not state the RIB contains any information whatsoever, such as perhaps the sequence number of the first frame to be retransmitted; the RIB is apparently just a signal for special action by a receiver.) When a receiver having no outstanding frames—and so not awaiting any retransmitted frames—receives such an RIB, it discards the next N frames and so avoids accepting duplicate frames. Such a receiver also acknowledges all the frames it discards, without having to actually determine the error condition of these frames. If instead the receiver requires some frame retransmissions, then some $0 < n < N$ frames must have already been discarded by the time the RIB arrives, so the receiver can know when to cease perfunctorily discarding frames and so when to restore normal attention to frames received. Thus simple counting rules determine how many frames to discard.

The tandem ARQ protocol proposed by Mase *et al.* assumes special supporting equipment aboard the satellite. It uses a point-to-point protocol for error control on the uplink between a transmitting earth station and the satellite, and a point-to-multipoint protocol for the downlink from the satellite to multiple receiving earth stations. In this scheme, a frame successfully received by the satellite from the transmitter on the Earth is immediately multicasted to the ground receivers. If the frame is unsuccessfully received by the satellite, then that frame is not multicasted to the receivers but a special dummy frame is sent instead. This dummy frame is discarded upon receipt by the receivers. The point-to-point ARQ between the transmitter and the satellite also uses

an RIB to maintain synchronization between the uplink and downlink transmissions. The tandem system is shown to have higher throughput than the end-to-end system due to a reduction of propagation delay for initiating retransmissions and a decoupling of uplink and downlink errors.

An inefficiency to be observed in this work is that each receiver must positively acknowledge all frames successfully received, even those frames already received successfully and retransmitted only to benefit other receivers. This requirement arises from the two aforementioned features of the “conventional” multicast ARQ protocol—which are effectively preserved in the proposed end-to-end scheme—and from the transmitter not “remembering” acknowledgements from receivers for a frame positively acknowledged by only some, but not all, receivers. If a history of such acknowledgements were to be maintained at the transmitter, and if the counting rules were to be replaced with the simple rule of checking if an arriving valid frame’s sequence number matches the number of the frame expected, the throughput could be improved, and the RIB could be eliminated.

1.2.3 Improved GBN Multicast ARQ

Gopal and Jaffe [7] perceived such inefficiencies and examined three GBN point-to-multipoint ARQ protocols. These three protocols differ in the degrees to which the transmitter regards past acknowledgements from receivers. In the first protocol, called *memoryless*, the transmitter ignores this history entirely. Hence this scheme has an “all-or-none” approach: all receivers must acknowledge the same transmission of a frame, or else the frame will be retransmitted. The extreme opposite approach is taken in the third protocol, called *full-memory*: the transmitter regards the entire available history of acknowledgements from all receivers for all frames transmitted

[within the ARQ window]. An intermediate approach is taken in the second protocol, called *limited memory*: full-memory operation applies for only the first packet in a group of consecutive packets retransmitted by the GBN protocol, but no history is maintained for any packets following that first one. Thus the transmitter considers acknowledgements of individual receivers but disregards the information of which receivers have acknowledged frames sent before the transmitter “goes back” for a retransmission. Not surprisingly, simulations establish the throughput superiority of the full-memory protocol over the other two protocols.

1.2.4 Selective Repeat Multicast ARQ

A selective-repeat (SR) version of Gopal and Jaffe’s examination of memoryless and full-memory protocols appeared in [8], by Sabnani and Schwartz. Instead of “memoryless” and “full-memory”, the nomenclature used in [8] is “fixed retransmission group (FRG)” and “dynamic retransmission group reduction (DRGR),” respectively. Paralleling the findings in [7], Sabnani and Schwartz found the DRGR technique yields superior throughput. They also found, again not surprisingly, that in a system with many receivers and high bandwidth-delay product broadcast channels, a GBN scheme does not provide usable throughput, while the SR version does.

1.2.5 Hybrid ARQ for Multicasting

The essential property to be gleaned from the works described above is that protocols which regard more of the history of ARQ acknowledgements, and retransmit fewer packets, tend to offer higher throughputs; this is hardly surprising. Having so increased the protocol complexity primarily at the transmitter, more recent literature has suggested adding sophistication to both the transmitter and the receivers. In particu-

lar, incorporating *hybrid ARQ* techniques—in which FEC and ARQ are combined—is the trend evident in multicast ARQ works of the past few years, reflected in the representative works cited below. This trend reflects similar developments in modern point-to-point ARQ schemes.

(The reader is cautioned that the term “hybrid ARQ” is a standard term in the literature and is not related to this dissertation’s term of “hybrid network” for a parallel arrangement of satellite and terrestrial networks.)

Deng proposed type-I hybrid GBN and SR ARQ schemes for multicasting in [9]. A shortened Hamming code is used for error correction while a cyclic redundancy check (CRC) is used for error detection.

A type-II hybrid SW ARQ scheme for multipoint transmission was proposed in [10]. In this scheme, rate-compatible BCH codes are used for error correction. Each time another retransmission for a particular packet is requested, the transmitter sends [additional] parity digits which, when combined with the original data packet, form a series of codewords of decreasing rate. After some number of unsuccessful attempts to recover the original data, the receiver requests the original data packet be retransmitted. (A similar modified type-II hybrid ARQ scheme for point-to-point communication was proposed in [11]).

A more sophisticated adaptive type-II hybrid GBN multicast ARQ scheme was considered in [12]. A concatenated coding scheme is used in this work. The outer code is a BCH code, and the inner code is a repetition code of rate $1/m$ ($m = 2, 3, \dots$). That is, BCH-encoded data is fed to a rate-1/2 convolutional encoder, which yields two sequences. The transmitted frame consists of $m - 1$ alternations of the two sequences obtained from the convolutional encoder. The inner convolutional code is used by the receiver for error detection, estimating channel BER, and deciding

whether to request a retransmission or to first attempt error correction with the outer BCH code (which is then also used for error detection). A frame's repetition number m depends on the window size, the number of receivers which have not positively acknowledged the frame, and the channel BER, and is found by dynamic programming techniques. The channel BER information is communicated from the receivers to the transmitter in the acknowledgements from the former. The transmitter maintains a list of which frames have been positively acknowledged, and which negatively, by all receivers. In the event the ARQ timer for a frame expires, the transmitter "goes back" to send that frame and all frames which follow.

1.2.6 Other Multicast ARQ Techniques

Multicast ARQ protocols not based on hybrid ARQ techniques have appeared in the literature as well. Wang and Silvester suggested sending multiple copies of each ARQ frame to improve the likelihood of correct reception before requesting a retransmission ([13]). The optimal number of frame repetitions for this scheme is determined by a dynamic programming technique. Another multiple-copies SR technique appeared in [14].

Jolfaei *et al.* suggested an SR ARQ protocol in which frames requiring retransmission are combined by modulo-2 addition into an "XOR frame" [15, 16, 17]. If the frames are properly combined, the XOR frame will contain at most one frame not received correctly by any particular receiver. Since the list of frames used to compose the XOR frame is sent with the XOR frame, an individual receiver can extract from the XOR frame the single constituent frame which that receiver has not yet correctly received.

1.2.7 Placement of this Dissertation

The works cited above representatively indicate the development of multicast ARQ may be characterized, although perhaps not precisely, as two generations. The first generation of multicast ARQ protocol development focused on adapting classical SW, GBN, and SR protocols for the multicast setting. After resolving fundamental issues, such as abandoning an “all-or-none” treatment of acknowledgements from the set of receivers, a second generation of multicast ARQ techniques arose. In this second (and continuing) generation, efforts have concentrated on hybrid ARQ techniques, perhaps fueled in part by the more-recent availability of hardware to support FEC at greatly diminished expense. Some non-hybrid ARQ multicasting techniques have also been proposed.

While better coding schemes may be introduced for multicast hybrid ARQ in the future, such advancements would likely be variations upon the hybrid ARQ theme which seems fairly mature in the literature. Accordingly, major advances in the multicast ARQ art are not to be expected from hybrid ARQ techniques. Some multicast ARQ schemes which do not incorporate hybrid ARQ techniques have been introduced, but these either have limited extensibility (in the case of “XOR” retransmissions) or rely on complex real-time or offline calculations to achieve optimal performance (in the case of multiple copies techniques). Accordingly, it seems likely that future novelty in multicast ARQ would likely entail methods substantially different from those so far considered in the literature.

This work presents such a different method: augmenting the broadcast satellite channel, which is ideally suited for simultaneous multipoint information distribution, with separate links to carry the retransmissions which have importance to individual receivers. The protocol chosen as the basis for this work comes from the end of

the first generation of multicast ARQ development: Sabnani and Schwartz's DRGR technique for SR multicast ARQ ([8]). This protocol is elegant in regarding acknowledgements intelligently but doesn't entail FEC complexities which are most beneficial primarily under high error conditions.

1.3 Overview of the Dissertation

The throughput achievable using ARQ error control for unicasting and multicasting, in pure-satellite and hybrid networks, is examined in Chapter 2. Chapter 3 discusses several details regarding implementing ARQ protocols, and considers how they affect throughput. Chapter 4 presents both simulation results and some additional discussion pertaining to multicasting. Conclusions and thoughts for future work are presented in Chapter 5.

Some helpful identities used in this work are presented in Appendix A. A glossary of notation is presented for reference in Appendix B.

Chapter 2

Throughput Analysis

In this chapter, the throughput of ARQ for unicasting and multicasting in pure-satellite and hybrid networks is calculated.

The ultimate purpose of an ARQ system is to deliver error-free information frames in proper order to a consuming process at the receiver (or receivers, in a multicast setting). Hence the performance measure is defined for this dissertation to be the throughput, v , calculated as the expected number of information bits released per second to the consuming process.

Let β be the expected number of frames sent by the transmitter per frame delivered to all receivers. With this definition, β is a measure of *inefficiency* (while its reciprocal is a measure of efficiency), and $\beta \geq 1$. This quantity will be used to calculate throughputs only in pure-satellite networks, although it will be used in a hybrid network setting for another purpose in the next chapter.

2.1 Point-to-Point Communication

2.1.1 Pure-Satellite Network

Unicasting in a pure-satellite network is examined first, and with the following assumptions and notational definitions:

1. An infinite supply of information frames is available for transmission, so the transmitter never idles for want of a fresh frame to send.
2. Unlimited buffering is available, and the ARQ window size is also unlimited; the SR ARQ protocol is then an “ideal” SR ARQ protocol.
3. All acknowledgements are delivered without errors.
4. The probability a frame sent via the satellite link arrives in error at the receiver is p_s .
5. An ARQ information frame comprises h header (overhead) bits and ℓ information bits (see Figure 2.1).
6. Acknowledgements are sent only for frames received without errors.
7. The satellite channel bit transmission rate is r_s (Figure 2.2).

It should be noted that the first of these assumptions is implicit in most selective-repeat ARQ throughput analyses although it is rarely mentioned. Also, the second assumption allows disregarding a finite transmission speed for acknowledgements.

The inefficiency measure, β , for a pure-satellite architecture with one receiver will be denoted as $\beta_{sat,1}$, and may be calculated as follows. With probability $1 - p_s$, some transmission of the frame eventually arrives successfully at the receiver. This may

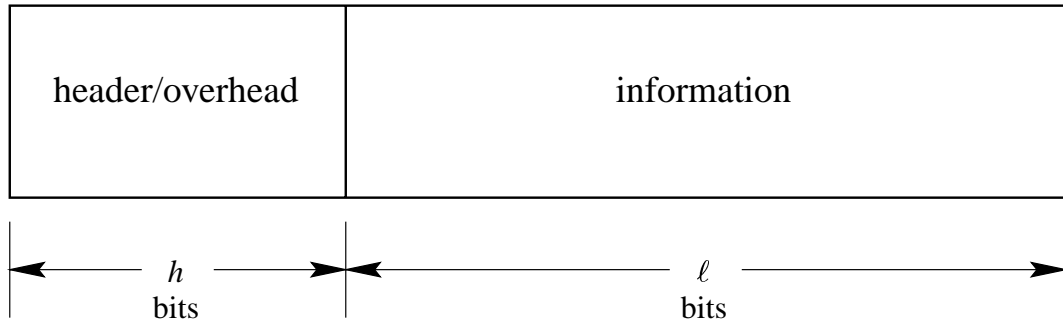


Figure 2.1: Information frame structure (as assumed for analysis).

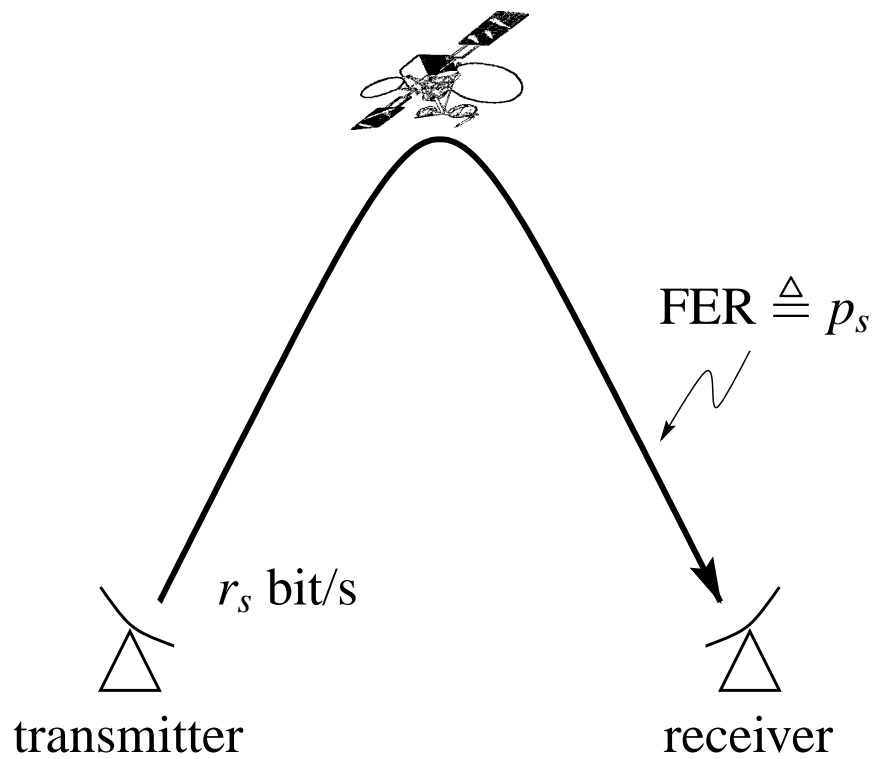


Figure 2.2: Information flow in a unicast pure-satellite network (FER: Frame Error Rate).

be preceded by zero or more unsuccessful transmissions, each with probability p_s .

Hence $\beta_{sat,1}$ is given by [1, 2]:

$$\beta_{sat,1} = \sum_{i=1}^{\infty} i(1-p_s)p_s^{i-1} = \frac{1}{1-p_s}. \quad (2.1)$$

The throughput in this setting, $v_{sat,1}$, is then

$$v_{sat,1} = \left(\frac{\ell}{\ell+h} \right) \frac{r_s}{\beta_{sat,1}} \quad (2.2)$$

$$= \left(\frac{\ell}{\ell+h} \right) (1-p_s) r_s. \quad (2.3)$$

2.1.2 Hybrid Network

For the hybrid network, the following assumptions are added (see also Figure 2.3):

1. The terrestrial link frame error rate is p_t .
2. The satellite channel bit transmission rate r_s exceeds r_t , the terrestrial channel bit transmission rate.
3. In the hybrid network, all retransmissions are sent terrestrially.

The protocol operation in the hybrid network may be modeled as the queueing system shown in Figure 2.4. Propagation delays are not regarded in this model since an unlimited window size and unlimited buffering have been assumed. In this model, information frames are sent continuously via satellite to the receiver. With probability $1-p_s$ a frame sent via the satellite link is successfully received. Hence the average rate at which frames are delivered successfully via the satellite link is the average frame flow rate at point “A” in the figure, $r_s(1-p_s)/(\ell+h)$ frames per second.

A frame which is corrupted in satellite transmission is queued at the transmitter in a terrestrial subsystem for retransmission. A retransmitted frame may be successfully

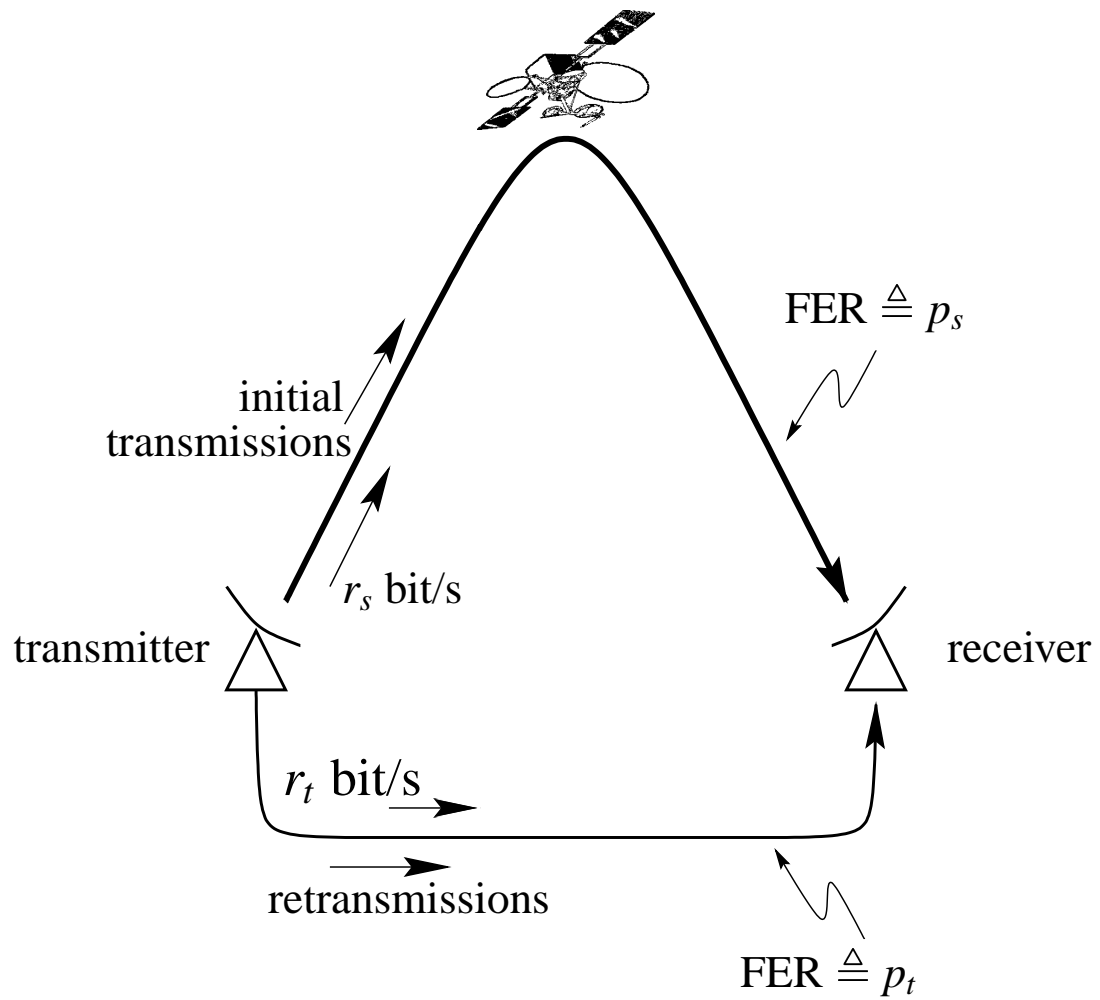


Figure 2.3: Information flow in a unicast hybrid network.

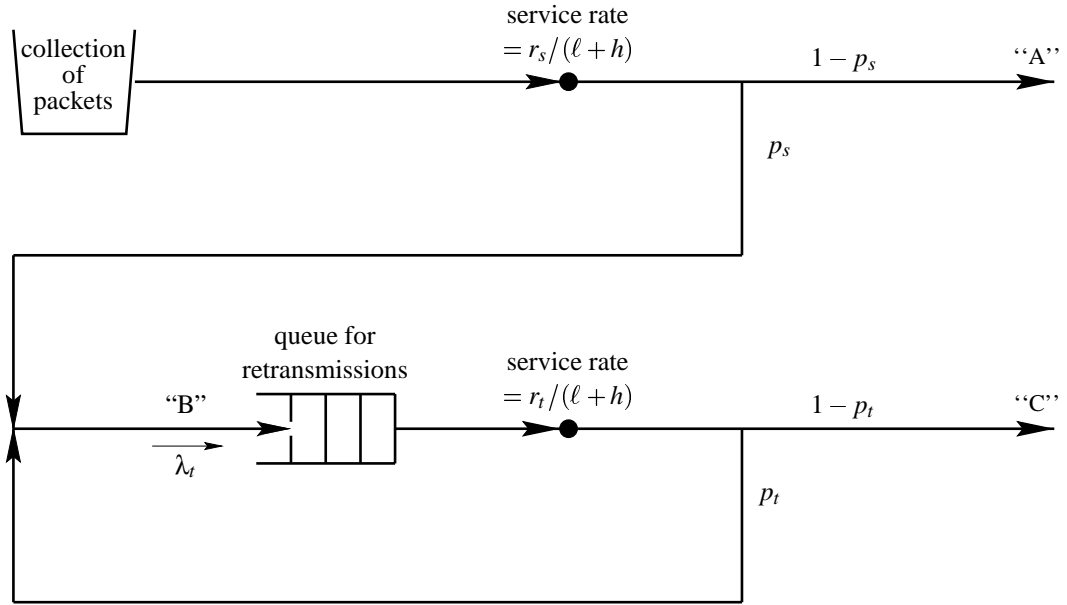


Figure 2.4: Packet flow model for the hybrid network.

received with probability $1 - p_t$, or the retransmission may be unsuccessful, in which case the frame is queued for another retransmission. Hence a frame may be retransmitted multiple times before it is successfully delivered. Let λ_t denote the average frame flow rate (in frames per second) at the input to the retransmissions queue, point “B” in the figure. The system will be said to be *stable* if the average input rate to the retransmissions queue is less than the rate at which retransmissions can be sent, i.e.

$$\lambda_t < r_t / (\ell + h), \quad (2.4)$$

and *unstable* otherwise.

Now if the system is stable, conservation of flow indicates $r_s p_s / (\ell + h) + \lambda_t p_t = \lambda_t$

or

$$\lambda_t = \frac{r_s}{\ell + h} \left(\frac{p_s}{1 - p_t} \right),$$

which in combination with (2.4) implies another expression of the stability condition,

$$\frac{r_s}{\ell + h} \left(\frac{p_s}{1 - p_t} \right) < \frac{r_t}{\ell + h}. \quad (2.5)$$

The corresponding average rate at which frames arrive successfully at the receiver via the terrestrial link is the average frame flow rate at point ‘‘C’’ in the figure, $\lambda_t(1 - p_t)$. If the system is unstable, then the flow rate at point ‘‘C’’ is limited to $r_t(1 - p_t)/(\ell + h)$. If v_C denotes the information throughput corresponding to this flow rate, then:

$$v_C = \begin{cases} \frac{\ell r_s}{\ell + h} p_s & , \text{ if stable} \\ \frac{\ell r_t}{\ell + h} (1 - p_t) & , \text{ if unstable} \end{cases}$$

Since the throughput rate at point ‘‘A’’ is $\ell r_s p_s / (\ell + h)$ bits per second, the throughput in a single-receiver hybrid architecture, $v_{hyb,1}$, is:

$$v_{hyb,1} = \begin{cases} \frac{\ell r_s}{\ell + h} & , \text{ if stable} \\ \frac{\ell r_s}{\ell + h} (1 - p_s) + \frac{\ell r_t}{\ell + h} (1 - p_t) & , \text{ if unstable} \end{cases} \quad (2.6)$$

These results may be combined with (2.5) to yield the more compact expression

$$v_{1,hyb} = \frac{\ell}{\ell + h} \min\{r_s, r_s(1 - p_s) + r_t(1 - p_t)\}, \quad (2.7)$$

in which the first term in the minimization corresponds to stable operation and the second corresponds to unstable operation.

It should be noted that Figure 2.4 does not necessarily represent an *implementation* of a hybrid network. In particular, one might expect prolonged unstable operation would lead to overflow of a finite retransmissions buffer. However, if the system is implemented with a common window for frames sent on the satellite and terrestrial links, then it is possible to assure no overflow of frames from the retransmissions buffer during prolonged unstable operation. Even so, the window size cannot be unlimited in an

implementation, and so the flow of fresh information frames on the satellite link may have to be cyclically suspended and resumed to allow for many terrestrial retransmissions if the system is unstable in the sense described above. (Such intermittent information delivery was indeed observed in simulation efforts, as will be described in Chapter 4.)

So, theoretically “unstable” operation is actually sustainable in practice with a finite window size. Despite this observation, the classification of a hybrid network’s operation as theoretically “stable” or “unstable” is a useful one, and so these labels will be retained for subsequent discussion.

2.2 Point-to-Multipoint Communication

For analyzing multicast networks, the assumptions of the point-to-point analysis are preserved, and the following others are added:

1. There are $M > 1$ receivers. (The results presented here also apply for $M = 1$.)
2. The noise processes experienced by all receivers are independent and identical.
3. There is no competition among receivers for access to the acknowledgment channel.
4. The ARQ protocol operation is according to the Dynamic Retransmissions Group Reduction (DRGR) technique described in [8]. The essential feature of this multicast selective-repeat ARQ protocol is that the transmitter maintains a history of which stations have acknowledged which frames. Accordingly, if a positive acknowledgement from receiver $m \in \{1, 2, \dots, M\}$ has been received

for frame F , an acknowledgement is not required from m for any retransmissions of F which may be required for other receivers in the network.

The throughput of the multicast system is defined to be the average of the unicast throughputs to all M receivers.

2.2.1 Pure-Satellite Network

In the multicast pure-satellite network, the transmitter continuously sends frames via the satellite multicast channel to the M receivers. Each receiver generates and sends to the transmitter an acknowledgment for each error-free frame received. The transmitter retransmits a frame if one or more receivers so request through their acknowledgements.

If $p_s = 0$, then clearly the throughput is $\ell r_s / (\ell + h)$ bits per second. If $p_s = 1$, then the throughput is obviously zero. Since $p_s = 0$ and $p_s = 1$ are uninteresting cases, assume hereafter, with only slight loss of generality, that $0 < p_s < 1$.

Let $m_k \in \{0, 1, \dots, M\}$ denote the number of receivers which successfully receive a frame F after exactly $k \in \{1, 2, \dots\}$ multicast transmission attempts to deliver F . Also let $\gamma(j)$ denote the probability with which the frame F is successfully delivered to all M receivers with j or fewer transmissions, where $j \in \{0, 1, \dots\}$. With this definition, $\gamma(0) \equiv 0$. For $j > 0$, the probability $\gamma(j)$ may be found by counting all possible combinations of the number of transmissions required to deliver frame F to each of the M receivers, given F was transmitted j times. Observe that the probability exactly k transmissions are required to successfully deliver a frame to a particular receiver is $p_s^{k-1} (1 - p_s)$. Hence the probability that each of m_k receivers requires exactly k multicast transmissions to successfully receive the frame is $[p_s^{k-1} (1 - p_s)]^{m_k}$. Now, given that the frame was transmitted j times, and all M receivers received the frame in

the course of the j transmissions, m_k is defined for $k \in \{1, 2, \dots, j\}$, with $\sum_k m_k = M$.

Hence $\gamma(j)$ is given by

$$\gamma(j) = \sum_{\substack{m_1=0 \\ \vdots \\ \sum_{h=1}^j m_h=M}} \cdots \sum_{m_j=0}^M \left[\binom{M}{m_1, m_2, \dots, m_j} \prod_{k=1}^j [p_s^{k-1} (1-p_s)]^{m_k} \right] \quad (2.8)$$

where the multinomial coefficient is given by

$$\binom{M}{m_1, m_2, \dots, m_j} = \frac{M!}{m_1! m_2! \cdots m_j!}.$$

A simpler way to calculate $\gamma(j)$ is to consider the complement of the events of the destinations not receiving F after j transmissions. That is, the probability a receiver successfully receives a frame in j or fewer transmissions is $1 - p_s^j$, so for M receivers

$$\gamma(j) = (1 - p_s^j)^M. \quad (2.9)$$

By its definition, $\gamma(j)$ is the cumulative distribution function for the number of transmissions required to successfully deliver a frame to all receivers. Then $\beta_{sat, M}$, the expected number of frames sent per frame delivered to all M receivers in the pure-satellite network, may be calculated as:

$$\beta_{sat, M} = \sum_{j=1}^{\infty} j[\gamma(j) - \gamma(j-1)] \quad (2.10)$$

(It should be acknowledged that although (2.10) was obtained independently of prior work, the same result was later found to have been obtained by Sabnani in [18], though with somewhat less explanation than given here.)

With $\beta_{sat,M}$ so given, the throughput for multicasting in a pure-satellite network, $v_{sat,M}$, is

$$v_{sat,M} = \left(\frac{\ell}{\ell + h} \right) \frac{r_s}{\beta_{M,sat}}. \quad (2.11)$$

It can be shown that $\beta_{sat,M}$, as specified in (2.10), evaluated with $M = 1$ is equivalent to $\beta_{sat,1}$ as specified in (2.1), and so $v_{sat,M}$ (2.11) is consistent with $v_{sat,1}$ (2.3). Substituting (2.9) into (2.10) yields:

$$\begin{aligned} \beta_{sat,M}|_{M=1} &= \sum_{j=1}^{\infty} j [(1 - p_s^j) - (1 - p_s^{j-1})] \\ &= \sum_{j=1}^{\infty} j [p_s^{j-1} - p_s^j] \\ &= \frac{1}{(1 - p_s)^2} - \frac{p_s}{(1 - p_s)^2} \\ &= \frac{1}{1 - p_s} \\ &= \beta_{sat,1} \quad , \end{aligned} \quad (2.12)$$

where (2.12) is obtained by (A.4) and (A.3). Hence $v_{sat,M}|_{M=1} = v_{sat,1}$.

Remarks Concerning $\beta_{sat,M}$

Before advancing to consider the multicast hybrid network, some further remarks concerning the quantity $\beta_{sat,M}$ are merited.

First, (2.10) is not a closed-form expression and so is somewhat unattractive. Two

other expressions for $\beta_{sat,M}$ can be obtained from (2.9) and (2.10):

$$\begin{aligned}
\beta_{sat,M} &= \sum_{j=1}^{\infty} j \left\{ (1 - p_s^j)^M - (1 - p_s^{j-1})^M \right\} \\
&= \sum_{j=1}^{\infty} j \left\{ \sum_{m=0}^M \binom{M}{m} (-p_s^j)^m - \sum_{m=0}^M \binom{M}{m} (-p_s^{j-1})^m \right\} \\
&\quad \text{by the binomial formula (A.10),} \\
&= \sum_{j=1}^{\infty} j \left\{ \sum_{m=0}^M \binom{M}{m} (-1)^m [(p_s^m)^j - (p_s^m)^{j-1}] \right\} \\
&= \sum_{j=1}^{\infty} j \left\{ \sum_{m=0}^M \binom{M}{m} (-1)^m (p_s^m)^{j-1} [p_s^m - 1] \right\}
\end{aligned}$$

Observe that the $m = 0$ term of the inner summation equals zero, and interchange the order of summations to obtain:

$$\beta_{sat,M} = \sum_{m=1}^M \binom{M}{m} (-1)^m [p_s^m - 1] \sum_{j=1}^{\infty} j (p_s^m)^{j-1}$$

Recalling that $0 < p_s < 1$, apply (A.4) to obtain:

$$\begin{aligned}
\beta_{sat,M} &= \sum_{m=1}^M \binom{M}{m} (-1)^m \frac{(p_s^m - 1)}{(1 - p_s^m)^2} \\
&= \sum_{m=1}^M \binom{M}{m} \frac{(-1)^{m+1}}{1 - p_s^m}
\end{aligned} \tag{2.13}$$

Equation (2.13) may be manipulated to obtain yet another expression for $\beta_{sat,M}$:

$$\begin{aligned}
\beta_{sat,M} &= \sum_{m=1}^M \binom{M}{m} \frac{(-1)^{m+1}}{(1-p_s^m)} \\
&= \sum_{m=1}^M \binom{M}{m} (-1)^{m+1} \sum_{i=0}^{\infty} (p_s^m)^i \\
&= - \sum_{m=1}^M \sum_{i=0}^{\infty} \binom{M}{m} (-p_s^i)^m \\
&= - \sum_{i=0}^{\infty} \left[\sum_{m=0}^M \binom{M}{m} (-p_s^i)^m - 1 \right] \\
&\quad \text{where the } -1 \text{ is obtained by including an } m=0 \text{ term;} \\
&= \sum_{i=0}^{\infty} \left[1 - (1-p_s^i)^M \right] \\
&= 1 + \sum_{i=1}^{\infty} \left[1 - (1-p_s^i)^M \right] \tag{2.14}
\end{aligned}$$

Thus $\beta_{sat,M}$ can be calculated by three expressions:

$$\begin{aligned}
\beta_{sat,M} &= \sum_{j=1}^{\infty} j \left\{ (1-p_s^j)^M - (1-p_s^{j-1})^M \right\} \\
&= \sum_{m=1}^M \binom{M}{m} \frac{(-1)^{m+1}}{(1-p_s^m)} \\
&= 1 + \sum_{i=1}^{\infty} \left[1 - (1-p_s^i)^M \right]
\end{aligned}$$

Although the second of these expressions is perhaps the most elegant of the three, numerical testing indicates this expression is not necessarily the most practical. It was found that the first of these expressions—with truncation of the summation as will be described in an example at the end of this chapter—yielded the most accurate results. The problem with the second and third expressions was traced to quantities

of form $1 \pm x$ being inaccurately computed as equaling unity when x is sufficiently small, a consequence of finite precision arithmetic. The expression interpreter used for numerical testing was intelligently designed to avoid such problems where possible. Although the first expression indeed includes quantities of the form $1 \pm x$, the interpreter evidently parsed this expression better than it did the other two. A different calculation tool or method might favor the second or third expression.

A last remark concerning $\beta_{sat,M}$ regards how this quantity varies with M . Let $0 < a < b$ with $a, b \in \{1, 2, \dots\}$. Equation (2.14) will be used to show that $\beta_{sat,b} > \beta_{sat,a}$. Now, $0 < p_s < 1$, so $0 < (1 - p_s^i) < 1$ for any $i \in \{1, 2, \dots\}$. Then:

$$(1 - p_s^i)^b = (1 - p_s^i)^a (1 - p_s^i)^{b-a}$$

The last factor on the right is less than unity, so the following chain of implications is obtained:

$$\begin{aligned} (1 - p_s^i)^b &< (1 - p_s^i)^a \\ 1 - (1 - p_s^i)^b &> 1 - (1 - p_s^i)^a \\ 1 + \sum_{i=1}^{\infty} [1 - (1 - p_s^i)^b] &> 1 + \sum_{i=1}^{\infty} [1 - (1 - p_s^i)^a] \\ \beta_{sat,b} &> \beta_{sat,a} \end{aligned}$$

Thus the inefficiency measure $\beta_{sat,M}$, which is the average number of times a frame must be transmitted to achieve delivery to all receivers, strictly increases with the number of receivers in the network. This is consistent with intuition regarding network performance: more receivers imply more retransmissions. Note also that while $0 < p_s < 1$ was assumed in deriving (2.14), that expression is also correct if $p_s \in \{0, 1\}$. It is then seen that $\beta_{sat,b} \geq \beta_{sat,a}$ with equality if and only if $p_s \in \{0, 1\}$. Further, $\beta_{sat,1} = 1/(1 - p_s)$ indicating that, in general, $\beta_{sat,M} \geq 1/(1 - p_s)$. Again

consistent with intuition, this last result indicates that the throughput in a multicast network with the receivers experiencing independent but identical noise processes is upper-bounded by the throughput attainable if the network has but one of those receivers and the others have been eliminated.

2.2.2 Hybrid Network

Although a multicast transmitter in a hybrid network must keep track of more information and service more receivers with retransmissions than its unicast counterpart, the receivers in the multicast setting are the same as in the unicast setting. Hence $v_{sat,M}$, the throughput for multicasting in a hybrid network, is the same as for unicasting in the hybrid network:

$$\begin{aligned} v_{hyb,M} &= v_{hyb,1} \\ &= \frac{\ell}{\ell + h} \min\{r_s, r_s(1 - p_s) + r_t(1 - p_t)\} \end{aligned}$$

In particular, the throughput in the hybrid network is independent of the number of receivers in the network.

2.3 Numerical Examples

Numerical examples using the throughput expressions derived above are now presented. These examples were constructed using the following assumptions beyond those mentioned earlier:

1. Binary symmetric channel (BSC) models characterize the terrestrial channels and the logical satellite channels between the transmitter and each receiver. The crossover probability (bit-error rate, BER) from each receiver's perspective, is

q_s for all logical satellite channels and q_t for all terrestrial channels. (Consequently, $p_s = 1 - (1 - q_s)^{\ell+h}$ and $p_t = 1 - (1 - q_t)^{\ell+h}$.)

2. The channel bit transmission rates are $r_s = 1536000$ (corresponding to “T1”/“DS1” rate) and $r_t = 33600$ (corresponding to a dial-up modem rate) bits per second in the satellite and terrestrial channels, respectively.
3. There are $\ell = 1776$ information bits and $h = 32$ overhead bits in all ARQ information frames, whether sent via satellite or via a terrestrial link. (The value of h was chosen supposing the ARQ frame has a 16-bit sequence number and a 16-bit CRC for error detection. The value of ℓ was chosen to maximize the throughput in a point-to-point satellite network, which is the reference network for comparison purposes. This maximization is calculated by a straightforward differentiation method presented in [19] and which is reviewed in section 3.3.1.1. For this maximization, q_s was taken to be 10^{-5} , the median value of the satellite link BERs examined below.)
4. In calculating $\beta_{sat,M}$, the infinite summation of (2.10) is approximated by truncating the summation at the minimum j such that $\gamma(j) > 1 - 10^{-10}$. (This truncation is justified not only as a fair approximation, but also because, in an actual network, a station which requests excessive numbers of retransmissions for each packet would likely be recognized by the transmitter as suffering from excessive noise, and would accordingly be disconnected from the communication.)

Calculated throughput values for point-to-point communication are presented in Figure 2.5, which illustrates several points. First, the figure clearly indicates a reduction of throughput with increase in number of receivers in the pure-satellite network. Second, the throughput in the hybrid network is shown to meet or exceed that in the

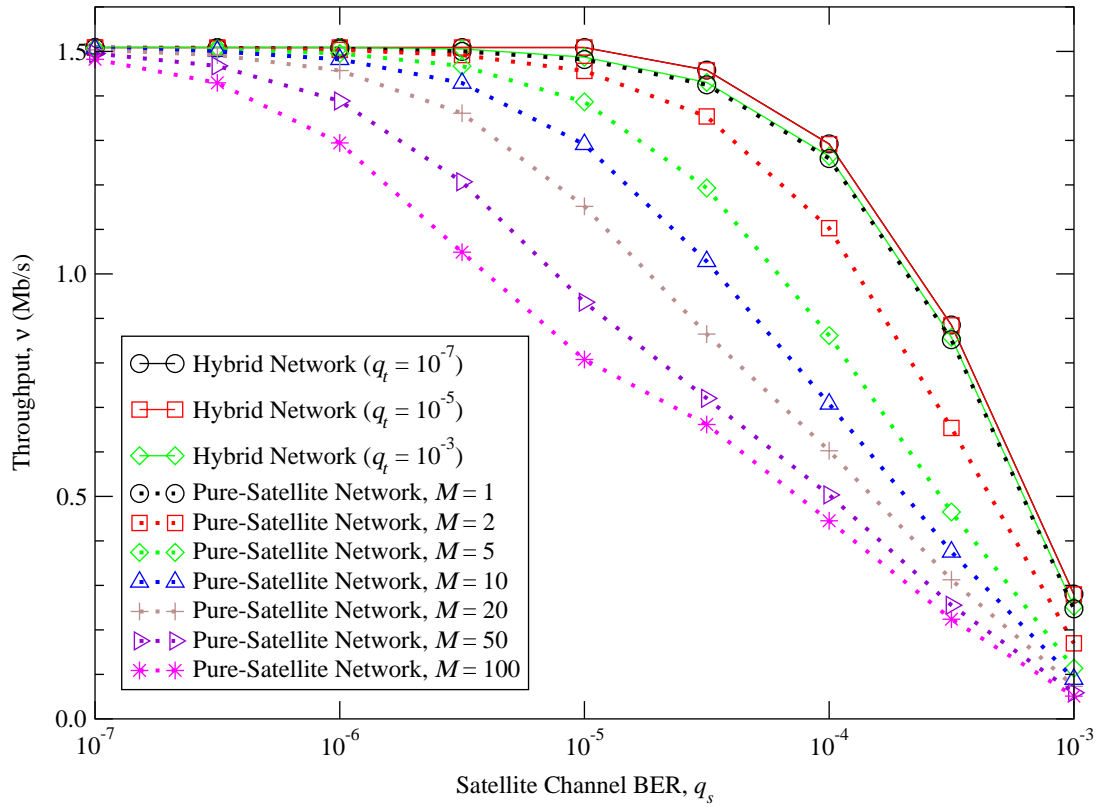


Figure 2.5: Throughput in point-to-multipoint networks.

pure-satellite network. Last, the hybrid network's throughput is independent of the number of receivers in the network.

Of course, achieving the higher throughput of a hybrid network requires terrestrial links from the transmitter to each receiver. This need for additional bandwidth prompts inquiry into the efficiency of the hybrid network. Several efficiency measures can be defined, on the basis of bits, packets, and frames. Fundamentally, the system user seeks to have his information delivered using the network. Segmenting data in packets or frames to achieve such delivery may be a known implementation element, but is largely irrelevant to the user, provided his data is delivered satisfactorily. Hence the numerator of an efficiency expression should be the network performance measure, the throughput. While the system operator knows that user data is segmented for transport, counting resources used in units of packets or frames does not reflect all the resources required to deliver the user's data, since frame overhead is not counted. Hence, for this dissertation, the throughput efficiency, η , is defined as the throughput divided by the sum of the information-carrying link bandwidths. That is, the efficiency in the pure-satellite network of M receivers is $\eta_{sat,M} = v_{sat,M}/r_s$, while the efficiency in a corresponding hybrid network is $\eta_{hyb,M} = v_{hyb,M}/(r_s + Mr_t)$.

Applying these definitions to the setting and results of the foregoing examples (using $q_t = 10^{-5}$ for the hybrid network) yields the results shown in Figure 2.6. The figure indicates efficiency (η) in the hybrid network decreases as the number of receivers (M) increases, a consequence of the efficiency measure's definition. The figure also indicates that, for a given number of receivers, the hybrid network is more efficient than the pure-satellite network if the satellite link BER exceeds some threshold value. A dashed curve drawn on the figure shows the approximate boundary between the satellite network being more efficient and the hybrid network being more efficient.

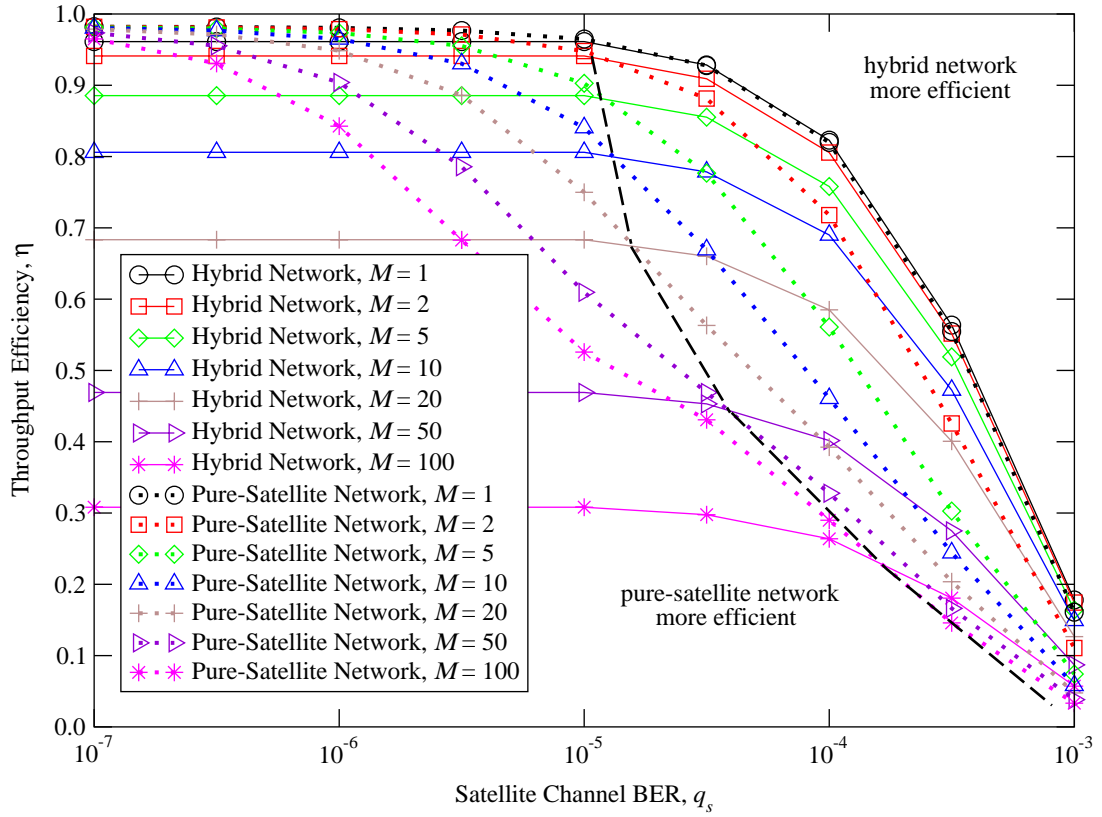


Figure 2.6: Throughput efficiency in point-to-multipoint networks. (For the hybrid network, $q_t = 10^{-5}$.)

So, while the hybrid network theoretically provides better throughput than the pure-satellite network, the former is more efficient only at higher satellite link BER values. This is true according to the efficiency measure defined above, but such efficiency calculations do not regard the multicast advantage provided by the different networks.

Chapter 3

Implementation Considerations

3.1 Chapter Overview

The previous chapter presented throughput analyses for ARQ operation in pure-satellite and hybrid networks, with one and many receivers. These analyses are approximate, for they ignore implementation issues such as finite window sizes. Such issues are important to examine because they have performance effects and because the protocols cannot be put to practice without considering them. This chapter describes in additional detail the protocols of interest in this work, and considers these implementation issues.

A goal of the work presented in this chapter was to identify appropriate settings for protocol parameters so that simulations of ARQ multicasting in pure-satellite and hybrid networks could be conducted. Although actual simulation results are deferred to the next chapter, this chapter remarks about the simulation software in a few instances. The simulator employed was the “ns” Network Simulator, version 2.1b5, hereafter referred to simply as “NS” ([20]).

3.2 Protocol Description

The multicast ARQ protocols examined in this dissertation are based upon the selective-repeat ARQ protocol with the “Dynamic Retransmissions Group Reduction (DRGR)” technique presented in [8]. To implement the protocol, international standards documents were consulted for guidance. The core logic of the SREJ protocol of ISO4335 ([21]) with the multi-selective reject option specified in ISO7776 ([22]) was selected as the point-to-point foundation for the protocol implemented. This foundation was chosen since it is similar to the protocols described in popular textbooks (e.g. [23, 24]) and because it is the core ARQ protocol used in many settings, such as X.25 transmission, and the Radio Link Protocol in the Global System for Mobile Communications (GSM, [25, 26]).

In this dissertation, the term *packet* will refer to a chunk of information to be delivered by the ARQ protocol from the transmitter to the receiver(s), and the term *frame* will refer to the ARQ construct, encapsulating a packet, which is used to achieve this delivery. Also, while the space of sequence numbers used in a practical ARQ protocol is necessarily finite, for clarity of presentation this fact will not be explicitly regarded in this chapter’s discussion.

In this protocol, the transmitting node sends ℓ -bit-long information packets in frames bearing a sequence number and a frame check sequence (FCS), such as a cyclic redundancy check (CRC). These two overhead fields have respective lengths of h_{seq} and h_{CRC} bits, respectively. The quantity h was defined in the previous chapter as the amount of overhead per frame, and so $h = h_{seq} + h_{CRC}$. The information frame structure is shown in Figure 3.1. For convenience, the total length of the frame will be denoted L , and so $L = h_{seq} + \ell + h_{CRC}$. The two notations of $\ell + h$ and L will be freely interchanged in this dissertation.

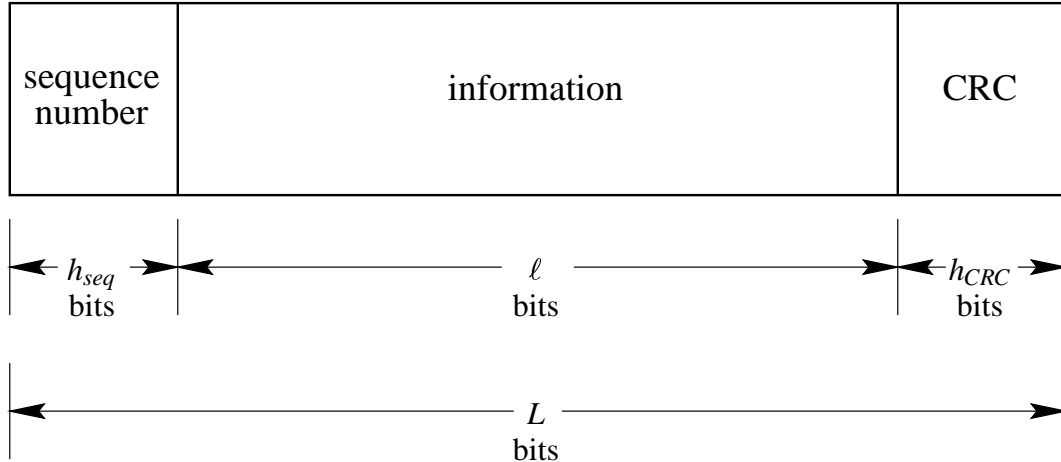


Figure 3.1: Information frame structure (as implemented).

The packets, and so the frames as well, are assumed to have a fixed length, and not variable lengths, since a throughput-maximizing packet length can be identified for a given amount of overhead. Also, there is no “arrival process” of information packets to the transmitter independent of the ARQ protocol operation. Rather, the transmitter sequentially fetches information packets from a source collection (which would typically be a large file) as and when the ARQ protocol can accommodate them. This fetching process is adopted instead of an autonomous arrival process for multiple reasons. First, it avoids confounding a protocol’s achievable performance, which is the interest of this dissertation, with a stochastic arrival process. Second, the performance results obtained for the fetched-packets system can serve to upper-bound those which would be obtained in an autonomous packet arrivals system. Last, the fetching behavior is characteristic of how SR ARQ protocols are commonly used, which is to deliver entire files, not to forward stochastically-arriving packets.

As each frame arrives at a receiver, the receiver uses the frame check sequence to verify the integrity of the frame. A frame found to contain any error is discarded without further action. If the frame is error-free, then the receiver generates an ac-

knowledge, to be described shortly. If the frame's sequence number is within the receiver's window of sequence numbers for frames to be accepted, the frame is stored in a buffer. If all earlier frames have been received successfully, and so their corresponding packets have been released to a higher-level consuming process, the information packet of the new frame is released as well. Receiver buffer space is reclaimed as frames are removed from the buffer and their corresponding packets are released to the higher-level consuming process.

The FCS protects not only the information packet, but the frame header (which comprises only a sequence number in this treatment) as well. Hence, if a frame is received with any error, then no part of the frame is useful—since the CRC does not indicate error positions—and so the entire frame is immediately discarded. The receiver's error detection process is assumed to be perfect: an error-free frame is never rejected, and an errored frame is never accepted. Not only is this a common assumption in the literature, it reflects the strength of CRCs in detecting errors in frames they protect. Although CRC-based algorithms can fail to detect some errors, the probability of such failures is so small that it is deemed negligible for the purposes of this work [2].

Each receiver generates an acknowledgement each time a frame is received without errors, even if the receiver has no need for that particular frame. The acknowledgement composition is shown in Figure 3.2. The acknowledgement bears a sequence number, called *RN*, which is the minimum sequence number such that all lesser-numbered frames have been successfully received by the receiver which generates the acknowledgement. This number is also the minimum sequence number within the receiver's window. Thus, the appearance of a particular value in the *RN* field communicates to the transmitter that all earlier frames have been successfully

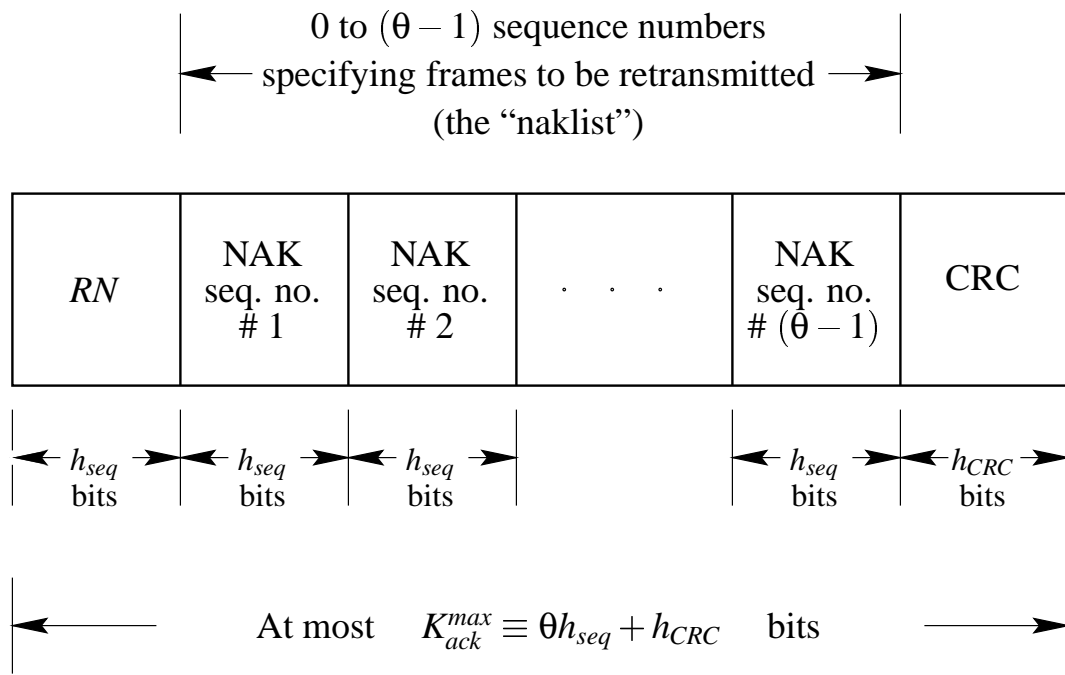


Figure 3.2: Acknowledgement composition.

received, and need not be retransmitted. Since the events of error-free arrival of a frame at the M receivers are assumed mutually independent, each receiver has its own RN value. This sequence number of course lies within the same space as the sequence numbers used to identify frames sent by the transmitter, and so is also h_{seq} bits long.

If the receiver has received successfully (without error) any frames beyond RN , then the acknowledgement also contains a field called the *naklist*. The naklist specifies, in increasing order, up to $\theta - 1$ sequence numbers for frames for which retransmission is required. If there are more than $\theta - 1$ such frames, the receiver specifies the earliest $\theta - 1$ of these. If there are fewer than $\theta - 1$ such frames, then the last number in the naklist is the least sequence number such that the frame it specifies, and all greater-numbered frames, were never received [correctly] by the receiver. (This number is what RN would be if the receiver were to require no retransmissions, as reckoned at the time the acknowledgement is composed.) Thus the RN field and the naklist together compose a list of strictly increasing sequence numbers. If two numbers appearing adjacently in this list differ by more than one, then all the skipped numbers between these two numbers are for frames which have been successfully received. In this fashion the naklist implicitly acknowledges positively those successfully-received frames bearing sequence numbers greater than RN .

The acknowledgement also bears a FCS, which protects the entire acknowledgement, and which is again assumed to be h_{CRC} bits long. With this description, an acknowledgement contains at most θ sequence numbers, and so has a maximum length of $K_{ack}^{max} \triangleq \theta h_{seq} + h_{CRC}$ bits.

Acknowledgements are sent to the transmitter on an *acknowledgements link* having bandwidth r_a bits per second. In the pure-satellite network, this is assumed to be a satellite link, while the acknowledgements link is assumed to be a terrestrial link in

the hybrid network.

For convenience, let the term *retransmissions link* refer to a link used for sending retransmissions. In the pure-satellite network, the retransmissions link is the satellite link, which is also used for the initial transmission of each frame. In the hybrid network, there is a terrestrial retransmissions link dedicated to each of the M receivers in the network.

Based on the acknowledgements received, the transmitter notes which frames have been successfully delivered to which receivers, and which must be retransmitted. The transmitter maintains a list of sequence numbers for frames to be retransmitted over each retransmissions link; this list will be called a *retransmissions list*. Hence there is a single such list in the pure-satellite network, and M such lists in the hybrid network. Each time after a frame is transmitted, the transmitter checks if any numbers are on the retransmissions list for that link. If so, the frame with the least number on the list is retransmitted, and that number is deleted from the list. Based on ARQ timer periods (these timers will be described shortly), which are the transmitter's best notions of round-trip times, the transmitter can decide if a retransmission request for a frame should be regarded, and so that frame's number should be added to the retransmissions list for the appropriate receiver, or the request should instead be ignored, since the frame has already been retransmitted but an acknowledgement for it has not had enough time to travel back to the transmitter.

Each time a frame is sent, either initially or for retransmission, a timer is started for the combination of that frame and each receiver from which the transmitter requires an acknowledgement for that frame. So, in a network with M receivers, M timers are started when a frame is sent for the first time (which is always a satellite transmission, in both the pure-satellite and hybrid networks). When a frame is positively acknowl-

edged by a receiver, the timer for that combination of frame and receiver is canceled. When a frame is retransmitted in the pure-satellite network, timers are started for that frame for receivers which have not yet acknowledged it. When a frame is retransmitted in the hybrid network, a timer is started for the combination of that frame and the single receiver to which the frame is retransmitted.

A timer expiration informs the transmitter that the associated frame was not successfully delivered to the associated receiver, and so that frame's number is added to the appropriate retransmissions list. The timer period will be discussed in section 3.3.2, below.

Various timing subtleties can be observed when the protocol operates. For example, it is possible that a frame number is added to the retransmissions list, but is acknowledged before the retransmission can be conducted. Such matters were carefully regarded in constructing the protocol simulations.

The transmitter ceases sending new frames if and when the transmitter's window is exhausted, which may never occur, depending on the window size and channel conditions. The window size, too, will be discussed below (section 3.3.3).

For simplicity, all links are assumed to be circuit-switched, and all acknowledgment links are assumed to be error-free.

3.3 Parameter Settings

This section considers how ARQ protocol parameters affect throughput, and seeks to find settings for these parameters so that throughput is maximized. This discussion also serves to describe further the operation of the protocols. The calculations developed in this discussion will be summarized at the end of this chapter, in section 3.3.5.

Before proceeding, it should be noted that although guidance for implementing the protocols was taken from ISO 7776, which has ISO 4335 as its foundation ([21, 22]), neither of these standards documents provides particular guidance for setting the parameters discussed below. This remark is offered not to denigrate the standards documents, but to reinforce the need for the work presented in this section.

3.3.1 Information Frame Length

For a link with a given noise process, the length of an ARQ frame sent on that link typically affects the probability the frame arrives intact at the destination. If a frame is too long, then the corresponding frame error rate (FER) will be large, and so throughput will be poor. If a frame is too short, then the FER may be small but the throughput may again be poor since overhead will constitute a large portion of the frame. Hence it is important to identify some intermediate value of frame length which will maximize throughput.

Further, it will be seen below that the FER, which is affected by frame length as just described, affects the setting of all ARQ protocol parameters. Accordingly the frame length is an operational parameter which affects all others.

For a given network, a change in frame length can change the number of frames which “fill” a link, or which remain unacknowledged. This affects the size of the sequence number space required (and so the size of the sequence number field). However, this second-order effect is not specifically regarded in this work, and a constant-size sequence number field of h_{seq} bits is assumed. Assuming a constant amount of overhead reduces the problem of determining the optimal frame length to determining the optimal *packet* length to be carried in a frame.

As in the numerical examples presented at the end of the previous chapter, bi-

nary symmetric channel (BSC) models are assumed to characterize the satellite and terrestrial channels between the transmitter and each receiver.

3.3.1.1 Pure-Satellite Network, Single Receiver

The throughput-optimal frame length for an ARQ protocol operating in a pure-satellite network with one receiver has been calculated by Schwartz in [19]. For completeness, as well as to motivate further discussion, this section reviews Schwartz's analysis (with some greater explanation).

The frame comprises h overhead bits and ℓ information bits. With h assumed constant, the problem of finding the optimal frame length is converted to finding the optimal ℓ . Let ℓ^* denote the value of ℓ which yields maximal throughput.

Let q_s denote the bit-error-rate (BER) of the satellite channel, and recall that $\ell + h$ is the frame length in bits. Assuming a BSC model, the frame-error-rate (FER) for the satellite channel, denoted as p_s , is

$$p_s = 1 - (1 - q_s)^{\ell+h}. \quad (3.1)$$

Substituting this expression into (2.3), the throughput in the single-receiver pure-satellite network is found to be

$$v_{sat,1} = \left(\frac{\ell}{\ell+h} \right) (1 - q_s)^{\ell+h}.$$

Differentiating with respect to ℓ yields:

$$\begin{aligned} \frac{\partial}{\partial \ell} v_{sat,1} &= \frac{\partial}{\partial \ell} \left[\frac{\ell}{\ell+h} (1 - q_s)^{\ell+h} \right] \\ &= (1 - q_s)^{\ell+h} \left[\frac{(\ell+h) - \ell}{(\ell+h)^2} \right] + \frac{\ell}{\ell+h} (1 - q_s)^{\ell+h} \ln(1 - q_s) \\ &= \frac{(1 - q_s)^{\ell+h}}{(\ell+h)^2} [h + \ell(\ell+h) \ln(1 - q_s)] \end{aligned}$$

To find ℓ^* , the above expression is set equal to zero and the strictly-positive factor outside the square brackets is dropped to obtain the quadratic equation

$$\ell^2 \ln(1 - q_s) + \ell h \ln(1 - q_s) + h = 0.$$

Applying the quadratic formula, and dropping an impossibly-negative value in the numerator, yields:

$$\begin{aligned} \ell^* &= \frac{-h \ln(1 - q_s) + \sqrt{h^2 \ln^2(1 - q_s) - 4 \ln(1 - q_s) h}}{2 \ln(1 - q_s)} \\ &= -\frac{h}{2} + \frac{1}{2} \sqrt{h^2 - \frac{4h}{\ln(1 - q_s)}} \\ &= \frac{h}{2} \left[\sqrt{1 - \frac{4}{h \ln(1 - q_s)}} - 1 \right] \end{aligned} \quad (3.2)$$

3.3.1.2 Hybrid Network, Single Receiver

As was described in the previous chapter, the hybrid network's operation can be either "stable" or "unstable," where the stability condition is as given in (2.5). Letting q_t denote the BER of the terrestrial channel, the corresponding FER under the BSC assumption is

$$p_t = 1 - (1 - q_t)^{\ell+h}. \quad (3.3)$$

Substituting into (2.5) yields

$$\frac{r_s}{\ell+h} \left(\frac{1 - (1 - q_s)^{\ell+h}}{(1 - q_t)^{\ell+h}} \right) < \frac{r_t}{\ell+h}$$

which simplifies to

$$r_s \left[1 - (1 - q_s)^{\ell+h} \right] < r_t (1 - q_t)^{\ell+h}. \quad (3.4)$$

Observe that as ℓ increases, the left-hand side (LHS) of this expression increases, while the right-hand side (RHS) decreases. So, as ℓ increases, the inequality becomes

untrue, and the operation in the hybrid network shifts from stable to unstable. This is to be expected, for as ℓ increases, the likelihood a frame sent via satellite will arrive in error also increases, implying a greater load is presented to the terrestrial subsystem in the hybrid network.

Note that for a given parameter set $\{h, r_s, r_t, q_s, q_t\}$, there may be no positive value of ℓ which satisfies (3.4), indicating that stable hybrid network operation is impossible. For example, if $\{h = 32, r_s = 1536000, r_t = 33600, q_s = q_t = 10^{-3}\}$, then evaluating (3.4) with these values, with $\ell = 0$, yields the untrue relation $48398 < 32541$. Since increasing ℓ increases the LHS but decreases the RHS of (3.4), it is clear that no positive value of ℓ can satisfy the inequality for the given parameter set.

Substituting for p_s and p_t in (2.6) yields:

$$v_{hyb,1} = \begin{cases} \frac{\ell r_s}{\ell+h} & , \text{ if stable} \\ \frac{\ell r_s}{\ell+h} (1 - q_s)^{\ell+h} + \frac{\ell r_t}{\ell+h} (1 - q_t)^{\ell+h} & , \text{ if unstable} \end{cases}$$

Differentiating with respect to ℓ yields:

$$\frac{\partial}{\partial \ell} v_{hyb,1} = \begin{cases} \frac{hr_s}{(\ell+h)^2} & , \text{ if stable} \\ \frac{hr_s}{(\ell+h)^2} (1 - q_s)^{\ell+h} + \frac{\ell r_s}{\ell+h} (1 - q_s)^{\ell+h} \ln(1 - q_s) \\ \quad + \frac{hr_t}{(\ell+h)^2} (1 - q_t)^{\ell+h} + \frac{\ell r_t}{\ell+h} (1 - q_t)^{\ell+h} \ln(1 - q_t) & , \text{ if unstable} \end{cases} \quad (3.5)$$

Observe that the derivative of $v_{hyb,1}$ is strictly positive if the hybrid network operation is stable. This suggests that, for stable operation, maximal throughput is obtained for the maximal value of ℓ for which stability is preserved. Let $\tilde{\ell}$ denote the positive value of ℓ which makes the LHS and RHS of (3.4) be equal. That is, for stable operation, throughput is maximized for the greatest positive integer $\ell \leq \tilde{\ell}$.

However, simulation work indicated that this procedure does not yield the optimal packet length for stable operation. The problem observed was the transmitter had to occasionally halt sending new frames because it had exhausted its window. Such an event can severely degrade throughput, particularly because of great propagation delays experienced with a satellite link. While window exhaustion may sometimes be alleviated with a larger window, simulations firmly established this approach does not appreciably help. Another way to alleviate window exhaustion, and which has a more direct performance benefit, is to adjust the packet length to reduce the need for retransmissions. That is, the average number of frames sent per frame delivered—a quantity denoted earlier as β —should be minimized. This minimization should be balanced against the consideration of how much frame header overhead is required. So, if $\beta_{hyb,M}$ denotes this inefficiency for a multicast hybrid network, then, paralleling the pure-satellite case, the appropriate optimization is to maximize $\ell/(\ell+h)\beta_{hyb,M}$.

Before calculating $\beta_{hyb,M}$ rigorously, this quantity can be found almost by inspection. Minimizing the number of frames sent per frame delivered involves maximizing the probability of successful delivery on the satellite link, which is $1 - p_s$, and the corresponding probability for the terrestrial link, $1 - p_t$. This maximization should not weight the two quantities evenly, since only a fraction of the traffic sent by satellite must be retransmitted terrestrially. Discussion in section 2.1.2 reveals this fraction is $p_s/(1 - p_t)$. So, the two quantities of $1 - p_s$ and $1 - p_t$ can be weighted accordingly, with a normalization factor, indicating

$$\frac{1}{\beta_{hyb,M}} = \frac{(1 - p_s) + \frac{p_s}{1 - p_t}(1 - p_t)}{1 + \frac{p_s}{1 - p_t}} = \frac{1 - p_t}{1 - p_t + p_s}.$$

This inefficiency measure can also be found more rigorously. A frame need be transmitted only once if the frame is successfully delivered via satellite; the probability of this event is $1 - p_s$. With probability p_s , terrestrial retransmissions will

be required for a frame. Each such retransmission can fail with probability p_t , until delivery is ultimately achieved. Therefore,

$$\begin{aligned}
\beta_{hyb,M} &= 1(1-p_s) + p_s [2(1-p_t) + 3p_t(1-p_t) + 4p_t^2(1-p_t) + \dots] \\
&= (1-p_s) + p_s \left[\sum_{i=2}^{\infty} i p_t^{i-2} (1-p_t) \right] \\
&= (1-p_s) + \frac{p_s}{p_t} \left[\sum_{i=1}^{\infty} i p_t^{i-1} (1-p_t) - (1-p_t) \right] \\
&= (1-p_s) + \frac{p_s}{p_t} \left[\frac{1}{1-p_t} - (1-p_t) \right] \tag{3.6} \\
&= 1-p_s + \frac{p_s}{p_t} \left[\frac{1}{1-p_t} - 1 \right] + \frac{p_s}{p_t} p_t \\
&= 1 + \frac{p_s}{p_t} \left[\frac{1}{1-p_t} - 1 \right] \\
&= 1 + \frac{p_s}{1-p_t} \\
&= \frac{1-p_t+p_s}{1-p_t},
\end{aligned}$$

where (3.6) was obtained using identity (A.4). Note that this yields the same expression for $\beta_{hyb,M}$ as the intuitive approach presented immediately above.

With $\beta_{hyb,M}$ in hand, the next step is to maximize $\ell/(\ell+h)\beta_{hyb,M}$, for which differentiation is indicated. To simplify the expressions, the derivative of $(\ell+h)\beta_{hyb,M}/\ell$ will be calculated; this will be just as useful, since minimizing $(\ell+h)\beta_{hyb,M}/\ell$ is equivalent to maximizing its reciprocal, which is the objective. Recalling (3.1), (3.3), and that $L = \ell + h$, differentiation yields:

$$\begin{aligned}
\frac{\partial}{\partial \ell} \left\{ \frac{\ell+h}{\ell} \beta_{hyb,M} \right\} &= \frac{\ell+h}{\ell} \frac{\partial}{\partial \ell} \beta_{hyb,M} + \beta_{hyb,M} \frac{\partial}{\partial \ell} \left\{ \frac{\ell+h}{\ell} \right\} \\
&= \frac{\ell+h}{\ell} \frac{\partial}{\partial \ell} \beta_{hyb,M} - \left(\frac{1-p_t+p_s}{1-p_t} \right) \frac{h}{\ell^2}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial}{\partial \ell} \beta_{hyb,M} &= \frac{\partial}{\partial \ell} \left\{ \frac{1-p_t+p_s}{1-p_t} \right\} \\
&= \frac{\partial}{\partial \ell} \left\{ \frac{(1-q_t)^{\ell+h} + 1 - (1-q_s)^{\ell+h}}{(1-q_t)^{\ell+h}} \right\} \\
&= \left\{ (1-p_t) \left[(1-q_t)^L \ln(1-q_t) - (1-q_s)^L \ln(1-q_s) \right] \right. \\
&\quad \left. - [1-p_t+p_s] (1-p_t) \ln(1-q_t) \right\} \times \frac{1}{(1-q_t)^{2L}} \\
&= (1-p_t) \left\{ (1-p_t) \ln(1-q_t) - (1-p_s) \ln(1-q_s) \right. \\
&\quad \left. - [1-p_t+p_s] \ln(1-q_t) \right\} \times \frac{1}{(1-p_t)^2} \\
&= \left\{ -p_s \ln(1-q_t) - (1-p_s) \ln(1-q_s) \right\} \times \frac{1}{(1-p_t)}
\end{aligned}$$

The final expression is then:

$$\begin{aligned}
\frac{\partial}{\partial \ell} \left\{ \frac{\ell+h}{\ell} \beta_{hyb,M} \right\} &= \left\{ -p_s \ln(1-q_t) - (1-p_s) \ln(1-q_s) \right\} \times \frac{\ell+h}{\ell(1-p_t)} \\
&\quad - \left(\frac{1-p_t+p_s}{1-p_t} \right) \frac{h}{\ell^2} \tag{3.7}
\end{aligned}$$

Finding the value of ℓ which sets this expression equal to zero—taking care that p_s and p_t are both functions of ℓ —yields the optimal value for ℓ for stable operation in the hybrid network.

One might be concerned that differentiating the throughput expression earlier did not yield the same result for the optimal packet length as just obtained. However, the throughput analysis of the preceding chapter assumed an unlimited window. Since the effect of window exhaustion is not reflected in the throughput calculation, it is

similarly not regarded if the packet length is determined exclusively by differentiating the throughput expression.

Notice that maximizing $\ell/(\ell + h)\beta_{hyb,M}$ was conducted without regard for stability. So, the value of ℓ obtained may correspond to either stable or unstable operation. Hence it would appear the optimal packet length for the hybrid network may *always* be found by determining the value of ℓ which sets (3.7) to zero. Once the optimal value of ℓ has been obtained, (3.4) may be consulted to determine if that value of ℓ yields stable or unstable operation.

However, simulations established the need to regard stability when calculating the optimal packet length. Results presented in the next chapter prove that unstable operation yields substantially inferior throughput, and once again the problem was traced to window exhaustion. Consider a given hybrid network in which ℓ is obtained as above, and (3.4) is violated, indicating unstable operation. When the network begins operating, frames are sent via the satellite link, and some fraction of them (p_s) do not arrive intact at the receiver. These frames are then presented to the terrestrial subsystem for retransmission. The network is “unstable” and so the average rate at which frames arrive at the terrestrial subsystem for retransmission exceeds the average rate at which frames are successfully delivered. The terrestrial subsystem quickly falls behind in processing the frames presented to it. The satellite transmitter obliviously continues sending frames at high speed, more frames arrive at the terrestrial subsystem, and the terrestrial subsystem falls further behind in processing the load of frames presented to it for retransmission. This continues until eventually the satellite transmitter sends the last frame of a window whose first frame is one being processed in the terrestrial subsystem. The satellite transmitter then idles, since it cannot send any new frames, and remains idle until the first frame in the window is successfully

delivered and acknowledged. The window then advances by one or more positions, allowing the satellite transmitter to send some number of frames until the window is again exhausted. The satellite transmitter then idles until the terrestrial subsystem achieves delivery of the first frame in the window. Thus, in the hybrid network, the satellite transmitter cycles between sending a burst of frames and then idling to wait for the terrestrial subsystem to catch up. During such idle periods the throughput is fairly low, since the satellite transmitter is silent and the terrestrial transmitter sends frames relatively slowly.

Note that increasing the window size does not change anything in the foregoing behavioral description except the length of the satellite transmitter's initial burst of continuous operation, which should be ignored when reckoning a long-term average measure such as throughput. Fundamentally, the throughput is limited because the terrestrial transmitter is slow and the satellite transmitter idles. The duty cycle of the satellite transmitter—the average fraction of time it is operating—is clearly related to p_s , p_t , r_s , and r_t , but is independent of the window size. Calculating the duty cycle was attempted but found to involve an intractable analysis of patterns of successful and unsuccessful delivery of frames on the two links. So, unlike in the pure-satellite network, an adverse performance consequence of finite window size that cannot be ameliorated by extending the window has been identified for the hybrid network.

If, though, the operation were stable, the idling of the satellite transmitter would be substantially reduced or eliminated. That is, disallowing unstable operation, and forcing ℓ to provide stable operation, may be expected to provide better throughput with a finite window size than if unstable operation is permitted. This is achieved by determining ℓ by maximizing $\ell/(\ell + h)\beta_{hyb,M}$ as described above, and then checking (3.4). If (3.4) indicates ℓ yields unstable operation, then ℓ should be set to $\lceil \tilde{\ell} \rceil$

(where $\tilde{\ell}$ was defined on page 43 to make the LHS and RHS of (3.4) equal). With ℓ computed in this latter way, the throughput may be expected to fall short of that theoretically achievable with ℓ computed exclusively by maximizing $\ell/(\ell+h)\beta_{hyb,M}$, but that theoretical throughput is not actually achievable in practice.

Both unstable-allowed and unstable-disallowed (stability-assured) operations are explored in simulation results presented in the next chapter.

3.3.1.3 Pure-Satellite Network, Multiple Receivers

For the pure satellite network having multiple receivers, it is more difficult to differentiate the throughput expression with respect to ℓ and to then numerically solve for the value of ℓ which sets the derivative to zero than to work with the throughput expression directly and numerically find the throughput-maximizing value for ℓ . That is, numerical evaluation and search can be employed to find

$$\ell^* = \arg \max_{\ell} \left\{ \left(\frac{\ell}{\ell+h} \right) \frac{1}{\sum_{j=1}^{\infty} j \left[\left(1 - p_s^j \right)^M - \left(1 - p_s^{j-1} \right)^M \right]} \right\}. \quad (3.8)$$

Such a direct approach was used for preparing the simulations for the next chapter.

3.3.1.4 Hybrid Network, Multiple Receivers

The number of receivers in a hybrid network affects throughput only by changing the probability that the window may be exhausted. Hence, the optimal packet length for the single-receiver hybrid network, as calculated above, is the optimal packet length for a hybrid network of any number of receivers.

3.3.2 Timer Periods

The next ARQ protocol parameter considered is the period for the retransmission timer. It will be eventually seen that this and the remaining ARQ protocol parameters—the window size and the maximum acknowledgement length—are all interrelated.

The timer forces retransmission of a frame if the frame is unacknowledged when the timer expires. In general, the purpose of this timer is to serve as a backup mechanism for forcing retransmissions in the event an acknowledgment requesting such a retransmission should be lost. The timer also serves to force retransmissions which may not be requested due to acknowledgments having limited size and so being unable to specify all frames requiring retransmission.

If the timer period is too small, then a timer will expire before an acknowledgment for the timer's associated frame can be received at the transmitter, and this will produce unnecessary retransmissions. If the timer period is too large, then appropriate timer-precipitated retransmissions will be unnecessarily delayed, which can harm throughput. Accordingly it is important to determine a proper intermediate value for the timer period.

3.3.2.1 Pure-Satellite Network

For simplicity, a single-receiver, pure-satellite network is considered first.

The timer period must be great enough to allow a frame to be transmitted, for the frame to propagate to the receiver, for an acknowledgement to be sent, and for the acknowledgement to be propagated back to the transmitter. Let the propagation delay from an earth station through the satellite and down to another earth station be τ_s seconds, and let the acknowledgements link have bandwidth r_a bits per second.

Noting that acknowledgements may have maximum length K_{ack}^{max} , it is seen that the timer period for ARQ operation in a pure-satellite network, T_{sat} , must satisfy

$$T_{sat} \geq \frac{L}{r_s} + \frac{K_{ack}^{max}}{r_a} + 2\tau_s. \quad (3.9)$$

Observe immediately that since $K_{ack}^{max} = \theta h_{seq} + h_{CRC}$, the timer period T_{sat} is related to the acknowledgement length parameter θ , which will be discussed below (section 3.3.4).

Conducting simulations with T_{sat} set to equal the RHS of (3.9) taught that such a timer period is in fact too small. If a frame arrives with error(s) at the receiver, the receiver does not seek to so inform the transmitter via an acknowledgement until the next error-free frame arrives. Hence, with T_{sat} set to equal the RHS of the expression above, the timer will likely expire before any acknowledgement requesting retransmission of an errored frame arrives at the transmitter. If the satellite channel is rather poor and the receiver generates maximal-length acknowledgements (K_{ack}^{max} bits long), the timer will certainly expire before the request arrives.

It is not immediately apparent why it should be preferable for the retransmission to be initiated by a request instead of by a timer expiration. After all, in the case just described, the timer expiration would elicit retransmission before the acknowledgement would. In general, however, if a positive acknowledgement should be lost or damaged in transit through the acknowledgements link, then the timer for the corresponding error-free frame being acknowledged will expire, forcing an unnecessary retransmission. So, it is preferable to initiate a retransmission in response to a request borne in an acknowledgement, although this may be somewhat later than retransmissions would be initiated by a timer expiration. Hence the timer period should be increased to allow for the occasional loss of acknowledgements.

Although this work assumes that the acknowledgements link is perfect, it is nonetheless proper to regard the possibility of acknowledgment link losses when setting T_{sat} . The solution to be described next can be easily extended to regard a case of an imperfect acknowledgments link. (For completeness, this extension will be briefly described near the end of this section.) Note, though, that even without link-corruption of acknowledgements, the setting of the timer period may be expected to affect throughput since there will likely be some frames retransmitted due to “legitimate” timer expirations.

The foregoing discussion indicates that T_{sat} must be increased beyond the RHS of (3.9). Since acknowledgements can be generated only when information frames arrive, it is sensible to formulate the necessary increase in T_{sat} as some positive integer number of frame inter-arrival periods. Assuming the transmitter’s window is not exhausted, the inter-arrival period for frames to the receiver is L/r_s seconds. Let $\phi_s \in \{2, 3, \dots\}$ be introduced as a variable to attend to this necessary increase, and so set

$$T_{sat} = \frac{\phi_s L}{r_s} + \tau_s + \frac{K_{ack}^{max}}{r_a} + \tau_s. \quad (3.10)$$

Thus a component of the timer period is the time required for ϕ_s consecutive frames to arrive at the receiver via the satellite link. With this formulation, the described increase with respect to the RHS of (3.9) has a magnitude of $\phi_s - 1$ frame inter-arrival periods, or $(\phi_s - 1)L/r_s$ seconds.

Now an appropriate value for ϕ_s must be found. This quantity will be set to force the probability of an undesirable event, to be immediately described, smaller than some arbitrary $\epsilon_{\phi_s} > 0$. Recall that ϕ_s was introduced to address the concern that non-generation of acknowledgements by the arrival of errored frames will effectively

delay notifying the transmitter that some errored frame must be retransmitted. Yet, it is preferable for retransmissions to be elicited by acknowledgements rather than timer expirations. So, the undesired event is that the timer expires before an acknowledgement requesting retransmission arrives at the transmitter. The probability of this event, for a given ϕ_s , will be denoted as $\Pr\{\text{expiration due to ack. nongeneration}|\phi_s\}$, and so ϕ_s will be sought such that $\Pr\{\text{expiration due to ack. nongeneration}|\phi_s\} \leq \varepsilon_{\phi_s}$ for a given $\varepsilon_{\phi_s} > 0$. In fact, the minimum such ϕ_s is preferred, so that the timer period is not overly generous, since too generous a timer period can have adverse effects as mentioned earlier.

To compute the probability of the described undesirable event, suppose some frame F_1 arrives errored at the receiver immediately after some frame F_0 arrives error-free. For a given ϕ_s , the timer for F_1 will then expire if more than $\phi_s - 1$ consecutive errored frames arrive immediately after frame F_1 . Hence:

$$\begin{aligned}
& \Pr\{\text{expiration due to ack. nongeneration}|\phi_s\} \\
& \approx \Pr\{\text{a frame arrives errored}\} \\
& \quad \times [1 - \Pr\{\leq \phi_s - 1 \text{ consecutive errored frames arrive} \\
& \quad \quad \quad \text{after the first errored frame}\}] \\
& = p_s \left[1 - \sum_{j=1}^{\phi_s-1} p_s^{j-1} (1 - p_s) \right]
\end{aligned}$$

where the summation index j counts errored frames after the initial errored frame (F_1 in the described scenario), and the $(1 - p_s)$ factor in the summation is the probability that the first frame after the j errored frames is error-free and so the run of consecutive errored frames terminates. It should be noted that the timer may expire even if fewer than $\phi_s - 1$ consecutive errored frames arrive immediately after F_1 since there may be many other frames, earlier than F_0 , which require retransmission. For this reason,

the above expression for $\Pr\{\text{expiration due to ack. nongeneration}|\phi_s\}$ is shown to be an approximation and not a strict equality. Although this case of timer expiration is a concern, it is a problem more closely allied to the setting for the maximum acknowledgement size, a parameter which is discussed below, and so this case is not regarded here further. Thus

$$\begin{aligned}
& \Pr\{\text{expiration due to ack. nongeneration}|\phi_s\} \\
& \approx p_s \left[1 - \sum_{j=1}^{\phi_s-1} p_s^{j-1} (1-p_s) \right] \\
& = p_s \left[1 - (1-p_s) \sum_{j=0}^{\phi_s-2} p_s^j \right] \\
& = p_s^{\phi_s}
\end{aligned}$$

by identity (A.2). Forcing $\Pr\{\text{expiration due to ack. nongeneration}|\phi_s\} \leq \epsilon_{\phi_s}$, while recognizing that ϕ_s must be at least two, yields

$$\phi_s = \max \left\{ \lceil \log_{p_s} \{ \epsilon_{\phi_s} \} \rceil, 2 \right\}. \quad (3.11)$$

One might wonder if this expression for ϕ_s is sufficient. In particular, one might wonder if the setting for ϕ_s should regard the possibility of acknowledgments being lost or dropped because the acknowledgements link has finite bandwidth, perhaps too little bandwidth to accept all acknowledgements as they are generated. To elaborate upon this concern, consider the receiver operation as depicted in Figure 3.3. The receiver ARQ protocol generates acknowledgements as described earlier, oblivious to whether the acknowledgements link has sufficient bandwidth to carry all the acknowledgements generated. If the satellite channel is perfect, acknowledgements will be generated at a rate of r_s/L acknowledgements per second, while the acknowledgements link can carry no more than $r_a / (h_{seq} + h_{CRC})$ minimal-length acknowledgements per second. Even under such ideal conditions, the acknowledge-

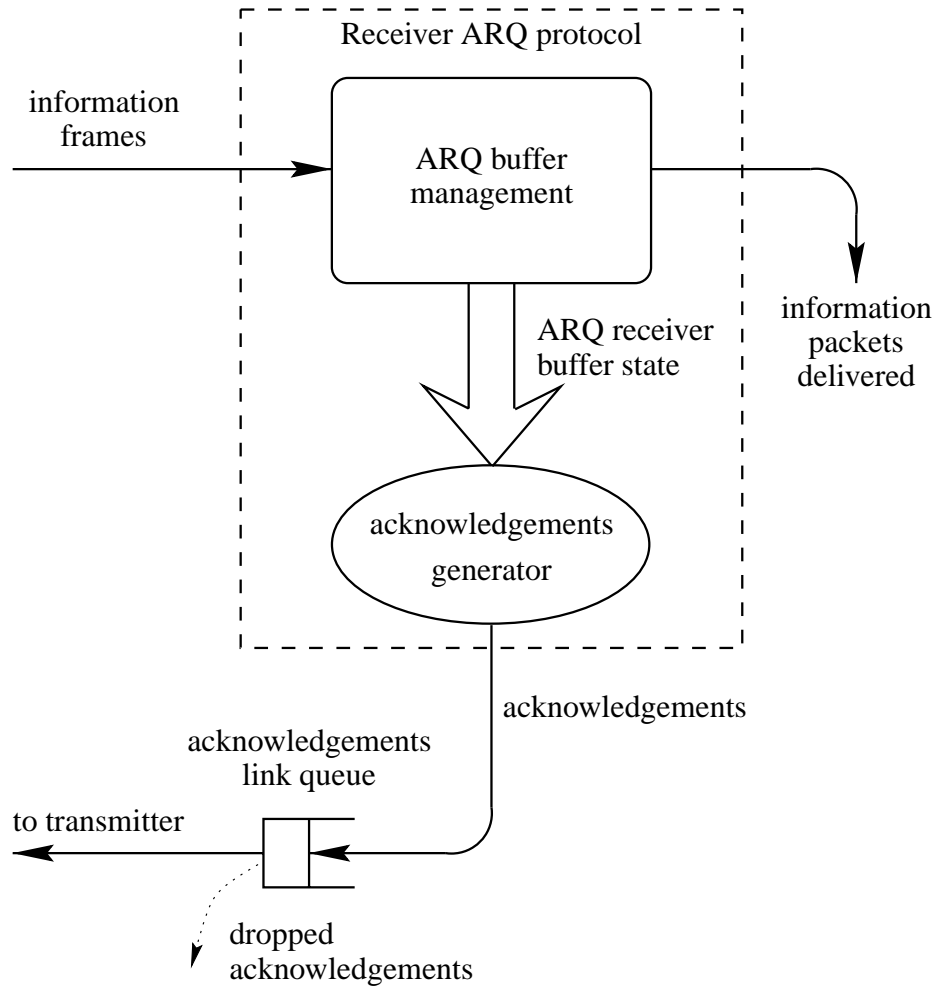


Figure 3.3: Acknowledgement generation and loss at the receiver.

ments link will not be able to accept all acknowledgements as they are generated if $(r_s/L) > r_a / (h_{seq} + h_{CRC})$. If this inequality is true, then acknowledgements will eventually have to be dropped, even if no frames actually require retransmission. With a poor satellite channel, maximal-length acknowledgements will be generated, and so some will have to be dropped if $(r_s/L) > (r_a/K_{ack}^{max})$. The figure reflects such possibilities as some acknowledgements being dropped from a small queue which buffers the acknowledgments supplied to the acknowledgements link. Therefore, if the acknowledgements link has a bandwidth constraint, acknowledgements can be lost after generation, and perhaps this, too, must be regarded in setting T_{sat} .

However, the construction of acknowledgements—in particular, the *RN* field—gives them a cumulative character. Also, the receiver’s ARQ buffer state is reflected no less well, and likely better, in later acknowledgements. Hence the acknowledgments link queue should drop older acknowledgements instead of newer ones. This can be achieved by requiring the link queue to drop acknowledgements from the *head* of the queue when full, instead of at the tail. Also, at most one acknowledgement should be queued so that a newer acknowledgement is not delayed by earlier, older acknowledgements which would be less valuable to the transmitter. These two acknowledgment link properties—dropping from the head of the queue, and allowing at most one acknowledgement to be queued—were both incorporated in the simulations conducted for this dissertation. (Specifically, in the NS simulation code, the “drop-front” option was set for the acknowledgements queue, and the “queue limit” for this queue was set to two to allow at most one acknowledgement to wait in the queue.) So, queueing losses of acknowledgements are indeed possible, but the combination of how acknowledgements are constructed and how they are dropped (when necessary) alleviates adverse effects of such losses. Hence no allowance for queueing loss of

acknowledgements is required when setting ϕ_s .

Observe, though, that if an acknowledgement is generated and *not* lost in the acknowledgement queue, then it may have to wait for completion of service for the most recent previous acknowledgement which was not dropped. In the worst case, these two acknowledgements are precipitated by consecutive information frame arrivals, the first acknowledgement enters service just before the second arrives, and at least the first of the two is K_{ack}^{max} bits long. Hence the second acknowledgement will have to wait in the acknowledgements queue for, at most, the full amount of time required to service the first acknowledgement. One might then wish to add K_{ack}^{max}/r_a to (3.10) and so set

$$T_{sat} = \frac{\phi_s L}{r_s} + \frac{2K_{ack}^{max}}{r_a} + 2\tau_s, \quad (3.12)$$

but the event of an acknowledgment waiting for service doesn't happen if, within a time of K_{ack}^{max}/r_a seconds, no acknowledgements are generated. So, an acknowledgment will not have to wait in the acknowledgments link queue for service if the previous $\lceil (r_s/L) (K_{ack}^{max}/r_a) \rceil$ consecutive information frames were all errored. The possibility of several consecutive information frames arriving errored and so not producing acknowledgements is the precisely the reason ϕ_s was introduced into T_{sat} , although another reason for ϕ_s is now seen: to account for possible queue waiting time. If ϕ_s as determined by (3.11) is greater than $\lceil (r_s/L) (K_{ack}^{max}/r_a) \rceil$, then there is no need for an additional K_{ack}^{max}/r_a to be included in T_{sat} as shown in (3.12). Even if (3.11) does not yield $\phi_s \geq \lceil (r_s/L) (K_{ack}^{max}/r_a) \rceil$, timer expiration due to non-generation of acknowledgements and timer expiration due to an acknowledgment being delayed while waiting for another to be serviced are not independent events. Hence adding another K_{ack}^{max}/r_a to T_{sat} as shown in (3.12) is too great an addition. Further, any addition should be incorporated in ϕ_s since acknowledgements are generated only in response

to frame arrivals. So, (3.11) should be adjusted to

$$\phi_s = \max \left\{ \lceil \log_{p_s} \{ \epsilon_{\phi_s} \} \rceil, \left\lceil \frac{r_s K_{ack}^{max}}{L r_a} \right\rceil, 2 \right\}, \quad (3.13)$$

to reflect the number of information frame inter-arrival periods corresponding to the time required to send a maximal-length acknowledgement.

Thus at last, for the pure-satellite network with a single receiver, the timer period is given by

$$T_{sat} = \frac{\phi_s L}{r_s} + \frac{K_{ack}^{max}}{r_a} + 2\tau_s \quad (3.14)$$

where, for a given ϵ_{ϕ_s} , ϕ_s is given by (3.13).

Before shifting to discuss timer periods in a hybrid network, a few small, additional matters should be addressed.

The first such matter is the case of an imperfect acknowledgements link. If the probability of a maximal-length acknowledgement being corrupted in the imperfect link is p_a , $0 < p_a < 1$, then the imperfection of the acknowledgments link can be regarded in the timer period by adding $\lceil \log_{p_a} \{ \epsilon_{\phi_s} \} \rceil$ as a fourth element to the set from which the maximum is found in (3.13).

Also, notice that the quantity K_{ack}^{max} appears in both (3.14) and (3.13), and again recall that $K_{ack}^{max} \equiv \theta h_{seq} + h_{CRC}$. Thus the timer period, both directly and through its parameter ϕ_s , depend on θ , which itself is an operating parameter.

Lastly, the foregoing discussion explicitly addressed only a single receiver in a pure-satellite network, but nowhere used the single-receiver assumption in obtaining the results given. Indeed, the same results apply for a multiple-receiver pure-satellite network.

3.3.2.2 Hybrid Network

In the hybrid network, two timer periods are required: one for frames sent via the satellite link, and one for frames retransmitted via the terrestrial link. Acknowledgements can be generated in such a network by the arrival of error-free frames by either link, which perhaps reduces the likelihood of no acknowledgements being generated during some given interval of time, as compared to the pure-satellite link. However, the timer periods should not be set only to regard average system characteristics, but they must also regard some rare worse-than-average situations. Since it is possible that at any particular instant either the satellite link or the terrestrial link is idle, the timer periods required for the hybrid network must account for such cases. That is, any degree by which having two links supplying frames to the receiver reduces the likelihood no acknowledgements will be generated during a particular interval cannot be exploited for reducing the timer periods.

So, the timer periods for the two links must be calculated independently. For this purpose, some given $\epsilon_{\phi_s} > 0$ and $\epsilon_{\phi_t} > 0$ will be assumed to be the limits imposed for upper-bounding undesirable timer expirations for frames sent on the satellite link and on the terrestrial link, respectively. For a given ϵ_{ϕ_s} and ϵ_{ϕ_t} , the corresponding ϕ_s and ϕ_t parametrize the degree to which the timer periods must be “extended” beyond a minimal value. Let T_{hyb}^s denote the timer period for frames sent on the satellite link in a hybrid network, and let T_{hyb}^t denote the timer period for frames sent on the terrestrial link. Since all acknowledgements are assumed to be sent on the terrestrial link, T_{hyb}^s differs slightly from T_{sat} in (3.14), namely

$$T_{hyb}^s = \frac{\phi_s L}{r_s} + \tau_s + \frac{K_{ack}^{max}}{r_a} + \tau_t \quad (3.15)$$

where ϕ_s is as given in (3.13) for the pure-satellite network,

$$\phi_s = \max \left\{ \lceil \log_{p_s} \{ \epsilon_{\phi_s} \} \rceil, \left\lceil \frac{r_s K_{ack}^{max}}{L r_a} \right\rceil, 2 \right\}. \quad (3.16)$$

Similarly, the timer period for frames sent terrestrially is

$$T_{hyb}^t = \frac{\phi_t L}{r_t} + \frac{K_{ack}^{max}}{r_a} + 2\tau_t \quad (3.17)$$

where

$$\phi_t = \max \left\{ \lceil \log_{p_t} \{ \epsilon_{\phi_t} \} \rceil, \left\lceil \frac{r_t K_{ack}^{max}}{L r_a} \right\rceil, 2 \right\}. \quad (3.18)$$

As in the earlier discussion of the timer period for a pure-satellite network, the fact there is only one receiver in the network was not used here, and so (3.15), (3.16), (3.17), and (3.18) apply for multi-receiver hybrid networks as well.

3.3.3 Window Size

The window size should be large enough to allow a frame to be initially transmitted as well as retransmitted some number of times, if need be, without the window being exhausted. The drawback of a huge window is principally the cost of additional memory required, and possible secondary costs such as faster computing hardware and a larger space (and larger field) for sequence numbering. On the other hand, if the window is rather small, then too few transmission attempts will be possible for a frame before the window is exhausted. Hence an intermediate value for the window size is needed.

To find the window size for ARQ operation in a pure-satellite network, N_{sat} , let $\omega \in \{2, 3, \dots\}$ denote a chosen maximum number of transmission attempts for a frame which are to be possible without exhausting the window. The value of ω is available to be set by the designer/implementer of an ARQ system.

It is worth explaining why ω is assumed to be an integer no less than two. With $\omega = 1$, the transmitter cannot send new frames while awaiting an acknowledgment for a retransmission, since the retransmission itself requires halting advance of the window. This is acceptable for a GBN protocol, and indeed $\omega = 1$ is an inherent characteristic of GBN ARQ, for which $\omega > 1$ cannot help. However, $\omega = 1$ greatly hampers a SR protocol, which is intended to allow transmission of new frames while awaiting acknowledgements for retransmissions. Indeed, if $\omega = 1$, the important benefit SR provides over GBN—the possibility of transmitting new frames while awaiting an acknowledgement for a retransmission—is lost. Consequently, the throughput in a point-to-point SR ARQ with $\omega = 1$ is not assured to exceed that of a corresponding GBN ARQ system. Hence $\omega \geq 2$ is required for SR ARQ operation.

Now, in textbook discussions of SR ARQ, the need for $\omega \geq 2$ is typically disguised in a statement that the sequence number modulus must be at least twice the number of frames which can be transmitted in one round-trip time (for example, see [23, 24]). Such a statement is true but it can be misleading. Since larger windows entail additional hardware costs, it is easy to interpret the textbook presentations as implying ω should be set equal to two. However, the textbook presentations do not elucidate that if $\omega = 2$, then if a third transmission (that is, a second retransmission) is required for a frame, then the window cannot advance while awaiting the acknowledgement for the third transmission. That is, although setting $\omega = 2$ allows for SR operation, better operation may be obtained if ω is greater than the minimally-required value. This work, then, puts the textbook assertion into a larger, more flexible context.

3.3.3.1 Pure-Satellite Network, Single Receiver

Based on the definition of ω , the window size should be the number of frames which can be sent within some multiple (ω) of the timer period. Strictly speaking, the timer period indeed exceeds the round-trip time. However, if the system is operating with poor channel conditions, then the margin ϕ_s provides T_{sat} will be exercised to a greater degree. If a timer is pending, then the transmitter should not sit idle simply because it hasn't yet received the corresponding acknowledgment, for otherwise a stop-and-wait operation will result. Of course, this permissiveness for the transmitter to send information frames must be bounded, and this bound is regarded with the parameter ω . Choosing to round upward to the nearest integer to allow one last frame to be sent before the timer expires for the ω -th transmission yields

$$N_{sat} = \left\lceil \omega \left(\frac{r_s}{L} \right) T_{sat} \right\rceil \quad (3.19)$$

with T_{sat} as given in (3.14).

Now an appropriate value for ω must be found. Since ω reflects how many times a frame can be sent without exhausting the window, it is sensible to find ω such that the probability of window exhaustion is less than or equal to some $\epsilon_\omega > 0$. In a pure-satellite network of but one receiver, the window will be exhausted if a frame is not successfully delivered after ω attempts. The probability of such a failure, resulting in window exhaustion, is p_s^ω . Forcing this probability to be no greater than ϵ_ω , and still requiring $\omega \geq 2$, yields

$$\omega = \max \left\{ \left\lceil \log_{p_s} \{ \epsilon_\omega \} \right\rceil, 2 \right\}. \quad (3.20)$$

3.3.3.2 Pure-Satellite Network, Multiple Receivers

For multiple receivers, the window size is still as given in (3.19), but any one of the M receivers can be the one responsible for the transmitter's window, which is common to all receivers, being exhausted. As mentioned in section 2.2.1, the probability a frame is delivered successfully to all receivers with ω or fewer transmissions is

$$\gamma(\omega) = (1 - p_s^\omega)^M .$$

So, the probability at least one of the M receivers requires an $(\omega + 1)$ -th transmission is $1 - \gamma(\omega)$. This probability should be no greater than ϵ_ω , indicating

$$\omega = \max \left\{ \left\lceil \log_{p_s} \left\{ 1 - (1 - \epsilon_\omega)^{1/M} \right\} \right\rceil, 2 \right\} . \quad (3.21)$$

Note that this result is consistent with (3.20).

3.3.3.3 Hybrid Network, Single Receiver

The window size for operation in the hybrid network, N_{hyb} , is calculated in a fashion similar to that for the pure-satellite network, accounting for the different paths used for initial transmission and retransmission:

$$N_{hyb} = \left\lceil \left(\frac{r_s}{L} \right) \left[T_{hyb}^s + (\omega - 1)T_{hyb}^t \right] \right\rceil \quad (3.22)$$

In this expression, both timer periods are multiplied by r_s/L , which is the amount of time required to send a frame by the satellite link, because the window advances only by sending new frames, which are always sent by satellite.

To use (3.22), the appropriate value of ω must be calculated for a given $\epsilon_\omega > 0$. For a given ω , the probability of window exhaustion with a single receiver is $p_s p_t^{\omega-1}$. Forcing this probability to be no greater than ϵ_ω , while still requiring $\omega \geq 2$, yields

$$\omega = \max \left\{ \left\lceil 1 + \log_{p_t} \left\{ \epsilon_\omega / p_s \right\} \right\rceil, 2 \right\} . \quad (3.23)$$

However, numerically evaluating this expression in the course of simulation work revealed a problem. In the situation of interest, q_s was varied while q_t was held fixed. Following the procedure described earlier, the packet length, ℓ , was obtained. As expected, ℓ was found to decrease as q_s was increased. For the particular quantities involved, the satellite channel frame error rate, p_s , corresponding to q_s and the associated ℓ , was found to increase with q_s . However, because q_t was fixed, and ℓ decreased with q_s , as a second-order matter p_t decreased as q_s increased. Applying (3.23) then produced ω -values which were a non-increasing function of q_s , and in fact would tend to decrease with q_s . That is, a smaller window size would be calculated for situations of poorer satellite channel performance. Now, by assumption, r_t is much smaller than r_s , so the round-trip time for a packet sent terrestrially is substantially larger than for one sent by satellite. If the window should be exhausted, much time may elapse before a timer at the transmitter expires to force another retransmission. As a consequence, the transmitter's window becoming exhausted can devastate throughput in the hybrid network, much more so than in a pure-satellite network. When the satellite channel is poorer, namely when q_s is greater, the throughput devastation is further exacerbated by there being so little successful delivery by satellite transmission.

So, while the analysis which produced (3.23) appeared sound, this expression yielded ω -values which could decrease with q_s , instead of values which should increase, and so the expression was insufficient for application in a practical setting. One might wish to develop an alternate method of finding ω which would regard the average duration of idle periods spent by satellite transmitter of the transmitting node when the window is exhausted. Unfortunately, calculating such a quantity amounts to computing the duty cycle of the satellite transmitter, which, as mentioned earlier (page 48), was found to be intractable. For lack of a better method, it was decided that

the calculation of ω for the hybrid network should not yield a result which is less than what would be obtained for the pure-satellite network. That is, (3.23) was changed to:

$$\omega = \max \left\{ \left\lceil 1 + \log_{p_t} \left\{ \epsilon_\omega / p_s \right\} \right\rceil, \left\lceil \log_{p_s} \left\{ \epsilon_\omega \right\} \right\rceil, 2 \right\} \quad (3.24)$$

Although this change does not assure that ω in a numerical scenario as just described is non-decreasing, it does particularly boost the value for situations of high satellite channel bit error rates, mitigating throughput deterioration in such conditions.

3.3.3.4 Hybrid Network, Multiple Receivers

For a given ω , the probability of window exhaustion in a hybrid network with M receivers is $1 - \left(1 - p_s p_t^{\omega-1}\right)^M$. Forcing this probability to be no greater than ϵ_ω , and still requiring $\omega \geq 2$, yields

$$\omega = \max \left\{ \left\lceil 1 + \log_{p_t} \left\{ \frac{1}{p_s} \left[1 - (1 - \epsilon_\omega)^{1/M} \right] \right\} \right\rceil, 2 \right\}.$$

Incorporating the desire that ω should be no less than in the pure-satellite network yields:

$$\omega = \max \left\{ \left\lceil 1 + \log_{p_t} \left\{ \frac{1}{p_s} \left[1 - (1 - \epsilon_\omega)^{1/M} \right] \right\} \right\rceil, \left\lceil \log_{p_s} \left\{ 1 - (1 - \epsilon_\omega)^{1/M} \right\} \right\rceil, 2 \right\} \quad (3.25)$$

This expression is consistent with its counterpart for the single-receiver case, (3.24).

The throughput in a hybrid network was stated in Chapter 2 to be insensitive to M under the assumptions given in that chapter. A finite window size violates one of those assumptions. It is seen here that the probability of window exhaustion depends on M . Consequently, one would expect the throughput in a multi-receiver hybrid network to also display sensitivity to M .

3.3.4 Maximum Acknowledgement Length

3.3.4.1 Overview

Ideally, acknowledgements should allow to the transmitter to know completely and instantly the receiver's ARQ frame buffer state. Unfortunately, each positive-length acknowledgement requires a positive amount of time to send over an acknowledgement channel of finite bandwidth, and so the acknowledgements used in practice must necessarily fall short of this ideal.

The acknowledgement construction described in section 3.2 and in Figure 3.2 comprises the key elements of an acknowledgement as described in ISO 7776, which is established atop ISO 4335 ([21, 22]). Each acknowledgement may contain a maximum of $\theta - 1$ sequence numbers, in addition to RN , to specify to the transmitter frames which must be retransmitted. By assumption, $\theta \geq 2$ so that frames requiring retransmission can be individually specified, as required for SR ARQ. The standards documents just cited provide no particular guidance on how to set θ (or any other) parameter for the protocol. There is a statement in ISO 7776 that equipment implementing the standards should support a maximum information frame length, including headers, "of not less than 1080 bits," "in order to support universal DCE [digital communication equipment] operation." Although acknowledgement frames are the focus of present discussion, such frames with $\theta > 1$ are described in the standards as information frames, with special indication in the header that they are to be treated as acknowledgements. Hence this 1080-bit specification would apply to acknowledgements with $\theta > 1$. However, ISO 7776 describes two protocol versions, one with modulo-8 sequence numbering and 8-bit headers, and one with modulo-128 numbering and 16-bit headers, and the 1080-bit specification is clearly intended to apply

universally and independently of which of these versions is used. An earlier draft of ISO 7776 also described a protocol version with modulo- 2^{31} sequence numbering and 64-bit headers, and this version contained the identical 1080-bit specification. Clearly, this 1080-bit specification is promulgated independently of protocol version, link transmission rates, and link qualities.

Hence the appropriate value of θ is left to the designer of a system which uses the protocol. A small value of θ allows an acknowledgement to specify less information to the transmitter than a large value. A particular adverse consequence of a small θ -value was noticed during simulations with high satellite channel BER values: since not all the correctly-received frames beyond the greatest one specified in the naklist can be acknowledged to the transmitter, the timers for the unspecified, successfully-received frames expire, precipitating many unnecessary retransmissions. Thus the protocol has a fit of behaving like a GBN protocol and the throughput suffers greatly. For subsequent discussion, this behavior will be termed “go-back- N degeneracy.”

Now a larger value of θ can reduce the occurrence of go-back- N degeneracy, but long acknowledgements have disadvantages of their own. One of these is that long acknowledgements have longer transmission times, which implies longer round-trip times, and so larger timer periods and window sizes. This translates to increased implementation cost, but is perhaps tolerable. Also, while an acknowledgement is being sent by a receiver, additional information frames arrive at the receiver via the satellite link, and some portion of these (about p_s) arrive errored.

This last observation prompts wondering if, by increasing θ , one gains more in providing the transmitter a better description of the receiver’s ARQ buffer state than one loses in additional errored frame “arrivals.” One can assert that, for a given θ , the “loss” is the number of “new” errored frames which “arrive” while the acknowledg-

ment is sent, yielding approximately $p_s \times (r_s/L) \times (\theta h_{CRC} + h_{seq})/r_a$ errored information frame “arrivals” while sending a maximal-length acknowledgement. Even so, there is no clear way to compare the “gain” achieved using different values for θ . Even if such a conjectured “gain” expression could be devised, the approach of comparing “gain” and “loss” disregards the probabilistic dependence of consecutive acknowledgements. That is, after the receiver sends one acknowledgement, additional errored frames can “arrive” while a second acknowledgement is sent, although the second acknowledgement may be similar or identical to the first. Hence comparing “gain” and “loss” was abandoned as being an unhelpful method for finding an appropriate value of θ .

Another approach considered was similar to that taken for finding ω : define an undesirable event and then find the parameter value which forces the probability of this undesired event to be less than a given $\varepsilon > 0$. Since too small a value for θ yields unnecessary retransmissions precipitated by timer expirations, it is appropriate to define the undesirable event as a timer expiring for a frame which in fact does not require retransmission, but which was not acknowledged to the transmitter since the naklist was too short to implicitly acknowledge the frame (that is, θ was too small). Calculating the probability of such an event equates to calculating the probability of the receiver composing a maximal-length naklist for the last acknowledgement to reach the transmitter before the timer expires for a frame whose sequence number lies outside the range covered by that naklist. An attempt to approximately calculate this probability indicated that calculating how many sequentially-numbered frames beyond RN had arrived at the receiver in either errored or error-free condition was required. However, this, too, involved an intractable analysis of patterns of successful and unsuccessful frame deliveries, including frames before RN . Hence this analysis

path had to be abandoned.

So, although intuition presented above describes the merits and drawbacks of large and small values of θ , this did not yield insight into how θ should be determined. As a final resort, then, it was decided to try a range of values for θ in simulations. Before doing so with arbitrarily chosen values, a way was sought to relate such values to system parameters, even if only remotely, which is now described.

3.3.4.2 Pure-Satellite Network

For simplicity, a worst-case situation in a particularly poor implementation of the protocol in a pure-satellite network is considered (a hybrid network will be addressed below). Assume ω and ϕ_s are each assigned their minimum possible values, namely $\omega = 2$ and $\phi_s = 2$. Assume further that, for the window size calculation, $\theta = 2$ is used. These assumptions serve not only to aggravate system performance, and so tend toward devising a worst-case situation, they also allows separating calculation of T , and so also N , from θ . The window size for this case is then

$$N = \left\lceil 2 \left(\frac{r_s}{L} \right) \left[\frac{2L}{r_s} + \frac{2h_{seq} + h_{CRC}}{r_a} + 2\tau_s \right] \right\rceil \quad (3.26)$$

$$\approx 2n_{sat} \quad (3.27)$$

where for convenience n_{sat} is defined as

$$n_{sat} \triangleq \left\lceil \left(\frac{r_s}{L} \right) \left[\frac{2L}{r_s} + \frac{2h_{seq} + h_{CRC}}{r_a} + 2\tau_s \right] \right\rceil .$$

With this definition, n_{sat} is approximately the minimum number of frames corresponding to a round-trip time in a pure-satellite network. Then obtain θ as

$$\theta = \lceil \alpha_\theta n_{sat} \rceil , \quad (3.28)$$

where $\alpha_\theta > 0$. Here, α_θ is envisioned as being less than unity, although this is not strictly required.

3.3.4.3 Hybrid Network

The approach used for the pure-satellite network is used also for the hybrid network.

Assuming $\omega = 2$ and $\phi_s = \phi_t = 2$ yields

$$N = \left\lceil \left(\frac{r_s}{L} \right) \left[\frac{2L}{r_s} + \tau_s + \frac{2h_{seq} + h_{CRC}}{r_a} + \tau_t + \frac{2L}{r_t} + \tau_t + \frac{2h_{seq} + h_{CRC}}{r_a} + \tau_t \right] \right\rceil, \quad (3.29)$$

$$\approx n_{hyb}^s + n_{hyb}^t \quad (3.30)$$

with

$$n_{hyb}^s \triangleq \left\lceil \left(\frac{r_s}{L} \right) \left[\frac{2L}{r_s} + \tau_s + \frac{2h_{seq} + h_{CRC}}{r_a} + \tau_t \right] \right\rceil,$$

$$n_{hyb}^t \triangleq \left\lceil \left(\frac{r_s}{L} \right) \left[\frac{2L}{r_t} + \frac{2h_{seq} + h_{CRC}}{r_a} + 2\tau_t \right] \right\rceil.$$

Here, n_{hyb}^s is approximately the minimum number of frames corresponding to a round-trip-time for a frame sent via the satellite link, and n_{hyb}^t is the analogous quantity for the round-trip time for a frame sent terrestrially. In the pure-satellite case, θ was calculated as a sub-multiple of the number of frames which can be sent during one round-trip time. The present hybrid network setting has two round-trip times corresponding to two different round-trip paths. The round trip corresponding to the initial frame transmission is selected as a basis for computing θ in the hybrid network for two reasons. First, the initial transmission round-trip path is more similar to the round-trip path of the pure satellite case. Second, frames to be specified in a selective acknowledgment must be lost in an initial transmission before they can be lost

in a terrestrial retransmission. That is, the frames lost during retransmission(s) are a subset of the frames lost during initial transmission. So, loss in the satellite link during initial transmission is the root event which causes a frame to fail to arrive at the receiver, and so is the root event which would require specifying the frame in a sufficiently-long selective acknowledgement. Therefore, θ for the hybrid network is obtained by $\theta = \lceil \alpha_\theta n_{hyb}^s \rceil$, where again $\alpha_\theta > 0$.

3.3.5 Summary for Setting ARQ Protocol Parameters

A summary of the formulas and methods presented for calculating parameters for the pure-satellite network is given in Figures 3.4 and 3.5. An analogous summary for the hybrid network is presented in Figures 3.6 and 3.7.

Given: h_{seq} , h_{CRC} , M , r_s , q_s , τ_s , r_a , α_θ , ϵ_{ϕ_s} , and ϵ_ω

1. Packet Length: Define:

$$\begin{aligned} h &= h_{seq} + h_{CRC} \\ L &= \ell + h \\ p_s &= 1 - (1 - q_s)^L \end{aligned}$$

If $M = 1$, compute:

$$\ell^* = \frac{h}{2} \left[\sqrt{1 - \frac{4}{h \ln(1 - q_s)}} - 1 \right]$$

If $M > 1$, then compute:

$$\ell^* = \arg \max_{\ell} \left\{ \left(\frac{\ell}{\ell + h} \right) \frac{1}{\sum_{j=1}^{\infty} j \left[(1 - p_s^j)^M - (1 - p_s^{j-1})^M \right]} \right\}$$

Round ℓ^* to the nearest integer to obtain the operating value ℓ .

2. Frame Error Probability: With $L = \ell + h$, compute $p_s = 1 - (1 - q_s)^L$.
3. Maximum Acknowledgement Length: Compute

$$n_{sat} = \left\lceil \left(\frac{r_s}{L} \right) \left[\frac{2L}{r_s} + \frac{2h_{seq} + h_{CRC}}{r_a} + 2\tau_s \right] \right\rceil,$$

select $\alpha_\theta > 0$, and then obtain

$$\theta = \lceil \alpha_\theta n_{sat} \rceil.$$

(continued in Figure 3.5)

Figure 3.4: Parameter calculations for pure-satellite network, steps 1 to 3 (of 5).

(continued from Figure 3.4)

4. Timer Period: With θ obtained in the previous step, let $K_{ack}^{max} = \theta h_{seq} + h_{CRC}$. Compute

$$\phi_s = \max \left\{ \left\lceil \log_{p_s} \{ \epsilon_{\phi_s} \} \right\rceil, \left\lceil \frac{r_s K_{ack}^{max}}{L r_a} \right\rceil, 2 \right\}$$

to obtain:

$$T_{sat} = \frac{\phi_s L}{r_s} + \frac{K_{ack}^{max}}{r_a} + 2\tau_s$$

5. Window Size: Compute

$$\omega = \max \left\{ \left\lceil \log_{p_s} \left\{ 1 - (1 - \epsilon_{\omega})^{1/M} \right\} \right\rceil, 2 \right\}$$

to obtain:

$$N_{sat} = \left\lceil \omega \left(\frac{r_s}{L} \right) T_{sat} \right\rceil$$

Figure 3.5: Parameter calculations for pure-satellite network, steps 4 and 5 (of 5).

Given: h_{seq} , h_{CRC} , M , r_s , r_t , q_s , q_t , τ_s , τ_t , r_a , α_θ , ϵ_{ϕ_s} , ϵ_{ϕ_t} , and ϵ_ω

1. Packet Length: Define:

$$\begin{aligned} h &= h_{seq} + h_{CRC} \\ L &= \ell + h \\ p_s &= 1 - (1 - q_s)^L \\ p_t &= 1 - (1 - q_t)^L \end{aligned}$$

Find ℓ^* satisfying

$$\left. \frac{\partial}{\partial \ell} \left\{ \frac{\ell + h}{\ell} \beta_{hyb, M}(\ell) \right\} \right|_{\ell = \ell^*} \equiv 0$$

where

$$\begin{aligned} \frac{\partial}{\partial \ell} \left\{ \frac{\ell + h}{\ell} \beta_{hyb, M}(\ell) \right\} &= \left\{ \left[-p_s \ln(1 - q_t) - (1 - p_s) \ln(1 - q_s) \right] \right. \\ &\quad \left. \times \frac{\ell + h}{\ell(1 - p_t)} \right\} - \left(\frac{1 - p_t + p_s}{1 - p_t} \right) \frac{h}{\ell^2}. \end{aligned}$$

Round ℓ^* to the nearest integer to obtain the operating value ℓ . If unstable operation is allowable, continue with the next step. Otherwise, check if ℓ satisfies the stability condition

$$r_s \left[1 - (1 - q_s)^{\ell + h} \right] < r_t (1 - q_t)^{\ell + h}.$$

If ℓ satisfies this inequality, continue with the next step. Otherwise, set $\ell = \left\lceil \tilde{\ell} \right\rceil$, where $\tilde{\ell}$ is the value of ℓ which sets equal the right- and left-hand sides of the above inequality.

2. Frame Error Probabilities: Compute L , p_s , and p_t , as defined in the previous step, for the value of ℓ just obtained.

(continued in Figure 3.7)

Figure 3.6: Parameter calculations for hybrid network, steps 1 and 2 (of 5).

(continued from Figure 3.6)

3. Maximum Acknowledgement Length: Compute

$$n_{hyb}^s = \left\lceil \left(\frac{r_s}{L} \right) \left[\frac{2L}{r_s} + \tau_s + \frac{2h_{seq} + h_{CRC}}{r_a} + \tau_t \right] \right\rceil,$$

select $\alpha_\theta > 0$, and then obtain

$$\theta = \left\lceil \alpha_\theta n_{hyb}^s \right\rceil.$$

4. Timer Period: With θ obtained in the previous step, let $K_{ack}^{max} = \theta h_{seq} + h_{CRC}$. Compute

$$\begin{aligned} \phi_s &= \max \left\{ \left\lceil \log_{p_s} \{ \epsilon_{\phi_s} \} \right\rceil, \left\lceil \frac{r_s K_{ack}^{max}}{L r_a} \right\rceil, 2 \right\} \\ \phi_t &= \max \left\{ \left\lceil \log_{p_t} \{ \epsilon_{\phi_t} \} \right\rceil, \left\lceil \frac{r_t K_{ack}^{max}}{L r_a} \right\rceil, 2 \right\} \end{aligned}$$

to obtain:

$$\begin{aligned} T_{hyb}^s &= \frac{\phi_s L}{r_s} + \tau_s + \frac{K_{ack}^{max}}{r_a} + \tau_t \\ T_{hyb}^t &= \frac{\phi_t L}{r_t} + \frac{K_{ack}^{max}}{r_a} + 2\tau_t \end{aligned}$$

5. Window Size: Compute

$$\begin{aligned} \omega &= \max \left\{ \left\lceil 1 + \log_{p_t} \left\{ \frac{1}{p_s} \left[1 - (1 - \epsilon_\omega)^{1/M} \right] \right\} \right\rceil, \right. \\ &\quad \left. \left\lceil \log_{p_s} \left\{ 1 - (1 - \epsilon_\omega)^{1/M} \right\} \right\rceil, 2 \right\} \end{aligned}$$

to obtain:

$$N_{hyb} = \left\lceil \left(\frac{r_s}{L} \right) \left[T_{hyb}^s + (\omega - 1) T_{hyb}^t \right] \right\rceil$$

Figure 3.7: Parameter calculations for hybrid network, steps 3 to 5 (of 5).

Chapter 4

Results and Discussion

This chapter presents results of a numerical example using simulations constructed with the operating parameter expressions obtained in the previous chapter. The characteristics of multicast pure-satellite and hybrid networks are explored further in subsequent discussion supported by additional simulation results.

4.1 Numerical Example

4.1.1 Assumptions for Numerical Example

For the example, the values given in Table 4.1 are assumed. The selection of these values is now addressed.

The satellite is assumed to be in geostationary orbit. Such a satellite has an orbital altitude of 35786 km above the earth, yielding an earth station-satellite-earth station propagation delay of 238 to 284 ms, depending on earth station terrestrial locations relative to the satellite [27]. The value of 300 ms is assumed for τ_s to allow for delays through satellite hardware, although no specific processing aboard the satellite is assumed.

Table 4.1: Assumptions for Numerical Example

Quantity	Meaning	Assumed Value	Units
τ_s	propagation delay through satellite channel	0.3	seconds
τ_t	propagation delay through terrestrial channel	0.125	seconds
h_{CRC}	CRC length	16	bits
h_{seq}	sequence number length	24	bits
r_s	satellite channel bandwidth	1536000	bits/second
r_t	terrestrial channel bandwidth	33600	bits/second
r_a	acknowledgements link bandwidth	33600	bits/second

The terrestrial link propagation delay τ_t comprises 30 ms of genuine propagation delay and 95 ms of modem processing delay. (The 95 ms of processing delay may seem surprisingly large, but is in fact based upon a measurement, as described in [28], p. 45).

A 16-bit CRC is an ITU-standard CRC length, and indeed such a CRC is specified in ISO7776 and ISO4335 ([21], [22]), and so h_{CRC} is set to 16 bits.

It was determined early in this work that an 8-bit sequence number would not be large enough to allow simulations with the necessary window sizes numbering in the hundreds or thousands of frames. Later, when constructing the NS simulation, it was discovered that NS requires frames to be an integer number of bytes in length, or else simulation errors would arise. Since θ would be a variable parameter in the simulations, h_{seq} would have to be a multiple of 8 bits. The smallest such multiple

which was expected to yield a sequence number space of sufficient size led to h_{seq} being set to 16 bits. It was later found that a 16-bit h_{seq} field also did not provide a sufficiently large numbering space for several cases, so ultimately h_{seq} was set to 24 bits for all simulations.

To fairly compare ARQ multicasting performance in pure-satellite and hybrid networks, it is sensible to set the acknowledgements link bandwidth, r_a , to be the same in the two networks. Since acknowledgements are assumed to be sent in the hybrid network on a full-duplex symmetric link of bandwidth r_t bits per second, the same value is assumed for r_a in both pure-satellite and hybrid networks. The value of 33.6 kb/s was chosen for r_t since this is the highest ITU-standard modem speed for sending data over a common telephone line. (A common so-called 56 kb/s modem actually sends data at 33.6 kb/s; the “56 kb/s” speed refers to the maximum possible speed at which data might be received if the other party in the modem connection has a high-grade digital connection to the telephone company’s network. No such high-grade line is assumed here.)

The value of 1.536 Mb/s is assumed for the satellite link bandwidth, r_s , as this is the “T1”/“DS1” rate in the rate hierarchy of digital circuits, and is a popular speed for high-speed digital service.

In the following discussion, single-receiver networks are considered before multi-cast networks.

4.1.2 Single-Receiver Pure-Satellite Network

The optimal packet length for the pure-satellite network is obtained from (3.2), which yields the results shown in Figure 4.1. To suit simulation constraints, the example proceeds with these packet lengths rounded to their nearest positive multiples of eight.

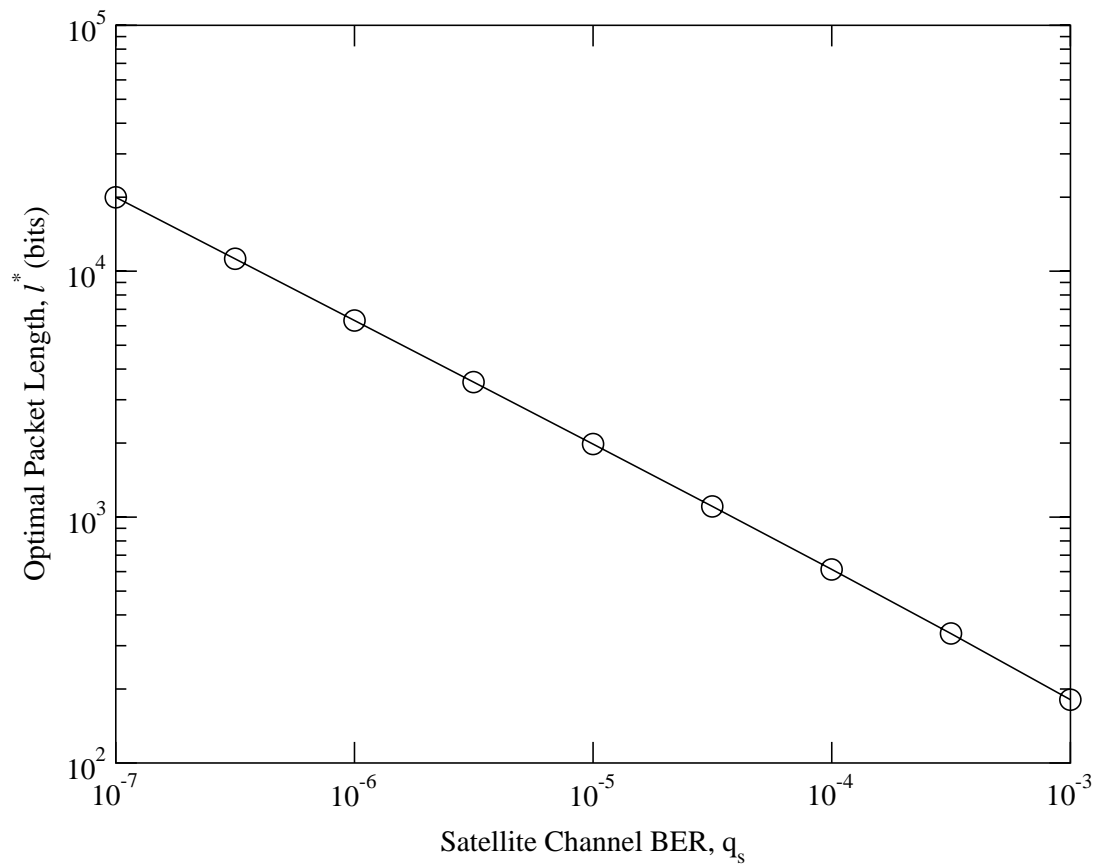


Figure 4.1: Optimal packet length for ARQ operation in the single-receiver pure-satellite network ($h = h_{seq} + h_{CRC} = 24 + 16 = 40$ bits).

According to Figure 3.4, after obtaining the appropriate value for ℓ , p_s should be found from (3.1) and then θ should be computed using (3.28). Following these steps yields the p_s -results shown in Figure 4.2, and the θ -results shown in Figure 4.3. The latter figure shows θ varies with q_s since θ depends on L , and L depends on q_s , as prescribed by Figure 3.4.

As discussed in the previous chapter, the ARQ parameters of θ , ω , and ϕ_s all may be expected to affect throughput and are inter-related. So, although the values of these parameters can be calculated independently according to the procedures in Figures 3.4 and 3.5, the network performance is affected by the ensemble of these parameters. Hence the effect each parameter exerts on throughput cannot be examined independently, but must be considered in the context of some given fixed set of the other parameters. Therefore, to see the throughput effects of θ , the ARQ parameters of ϕ_s and ω were calculated according to Figure 3.5 using $\varepsilon_{\phi_s} = 10^{-2}$ and $\varepsilon_{\omega} = 10^{-2}$, yielding the results shown in Figure 4.4. This figure indicates, that, in general, throughput improves with θ , particularly at higher BERs. However, it is somewhat peculiar that the curves are not everywhere concave as would be expected. This will be simply explained shortly, after presenting the simulation results showing the throughput effects of ω .

After obtaining θ , Figure 3.5 indicates that ϕ_s should be found. Several values for ε_{ϕ_s} were tried, beginning with 0.1, and decreasing to 10^{-12} , and ϕ_s was not found to vary with ε_{ϕ_s} . According to (3.13), this means ϕ_s was always set to $\lceil r_s K_{ack}^{max} / r_a L \rceil$, for the network of interest and the parameters considered. This indicates that allowing for a maximal-length acknowledgement to be completely transmitted was the determining consideration for setting ϕ_s in the network examined here. (The parameter ϕ_s was determined this way for the remainder of this single-receiver, pure-satellite network

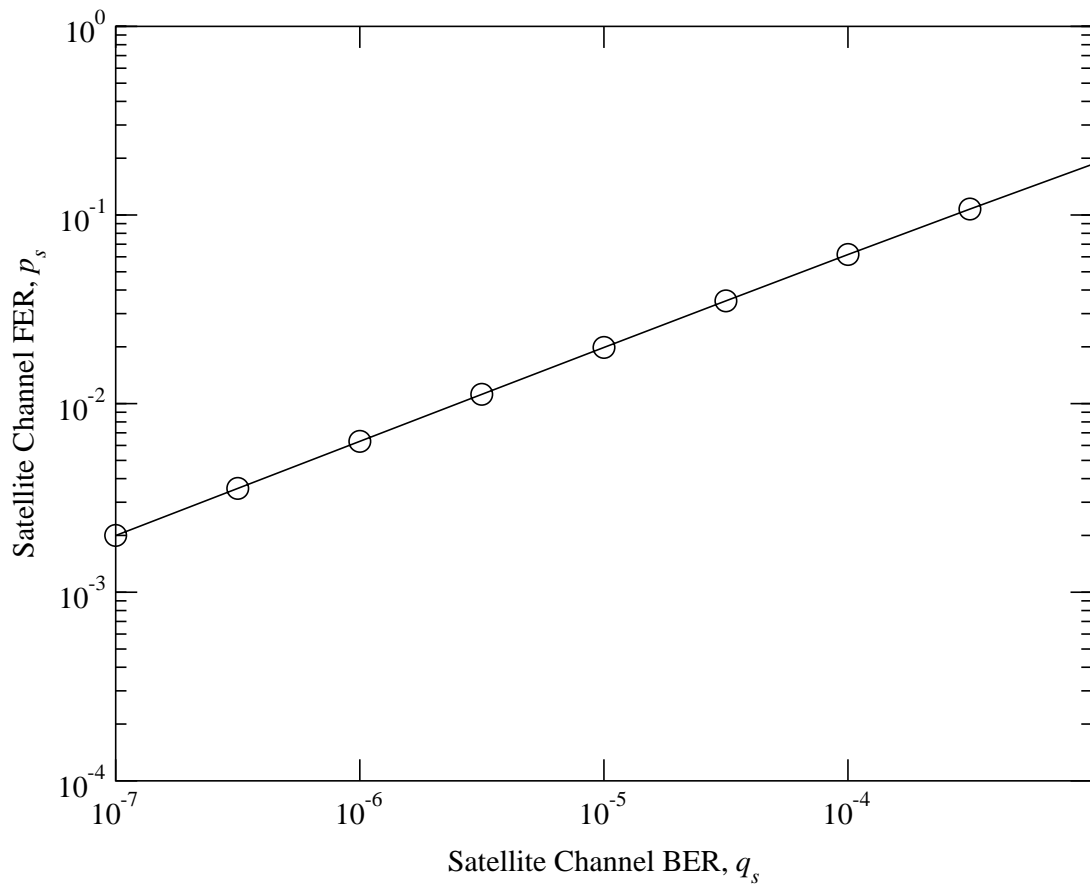


Figure 4.2: Satellite Channel FER, p_s , as calculated from optimal frame lengths.

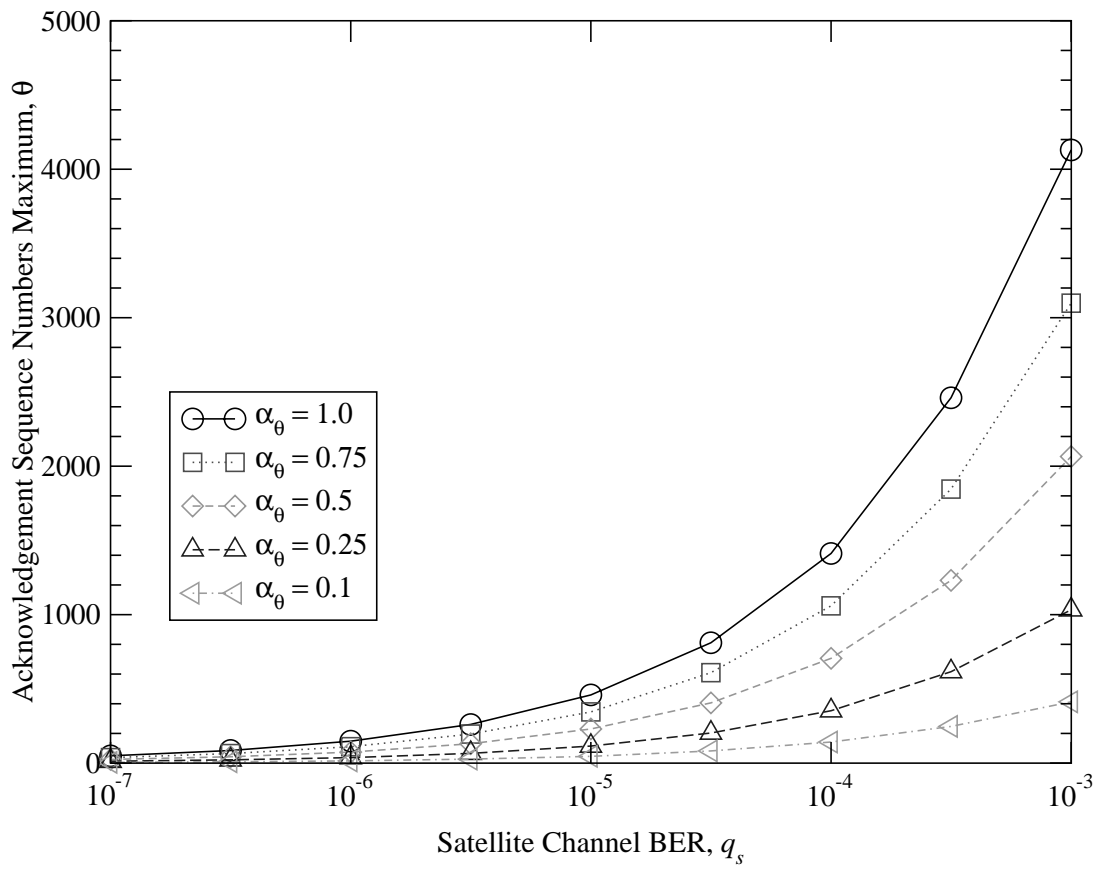


Figure 4.3: Acknowledgement sequence numbers maximum, θ , for the pure-satellite network.

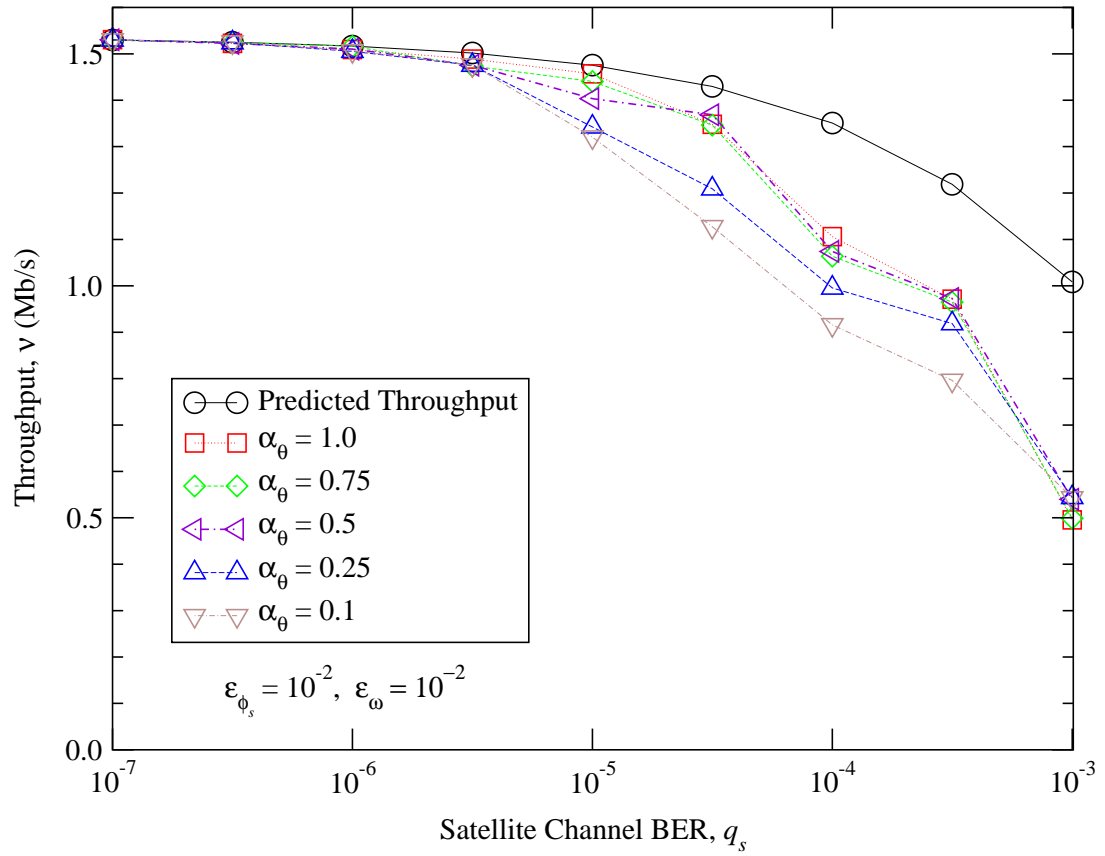


Figure 4.4: Throughput effects of θ in the pure-satellite network.

example.)

The next parameter considered was ω . Although Figure 4.4 indicates that increasing α_θ can improve throughput, it was decided to try $\alpha_\theta = 0.25$ and $\alpha_\theta = 0.1$ when considering the effects of different ω -values. This decision was made because ω has an explicit dependence upon M , the number of receivers, according to (3.20), while α_θ does not have such an explicit dependence. That is, it was expected that increasing ω would help throughput not only in a single-receiver network, but also in a multiple-receivers network (examined below), while increasing α_θ would not clearly offer a benefit for multicasting. The values so obtained for ω were the same for both $\alpha_\theta = 0.25$ and $\alpha_\theta = 0.1$ and are shown in Figure 4.5. The corresponding throughputs obtained by simulation are shown in Figure 4.6. This figure indicates that throughput improves as ω increases, as one would expect. The ostensibly odd dips in the curves of this figure, and the smaller ones in Figure 4.4, above, are easily attributed to jumps in the ω -values obtained for the given ϵ_ω -values, as shown in Figure 4.5. Note that Figure 4.6 indicates that the window size, as determined by ω for a given ϵ_ω , substantially affects throughput. Considering this figure against the background of figures presented earlier further indicates the window size parameter ω affects throughput more than do the other ARQ parameters. Although there is still significant disparity between predicted and simulated throughputs at $q_s = 10^{-3}$, it may be expected that decreasing ϵ_ω , and so increasing ω , might mitigate this shortfall. Also, at extremely high values of q_s , it is expected that the finite acknowledgement link bandwidth, r_a , and the finite acknowledgement length, K_{ack}^{max} , would allow the transmitter to know of only a small portion of the frames which require retransmission, and so would depress throughput below a theoretical value. However, having found parameter values that yield simulated throughputs which closely match the predicted throughputs

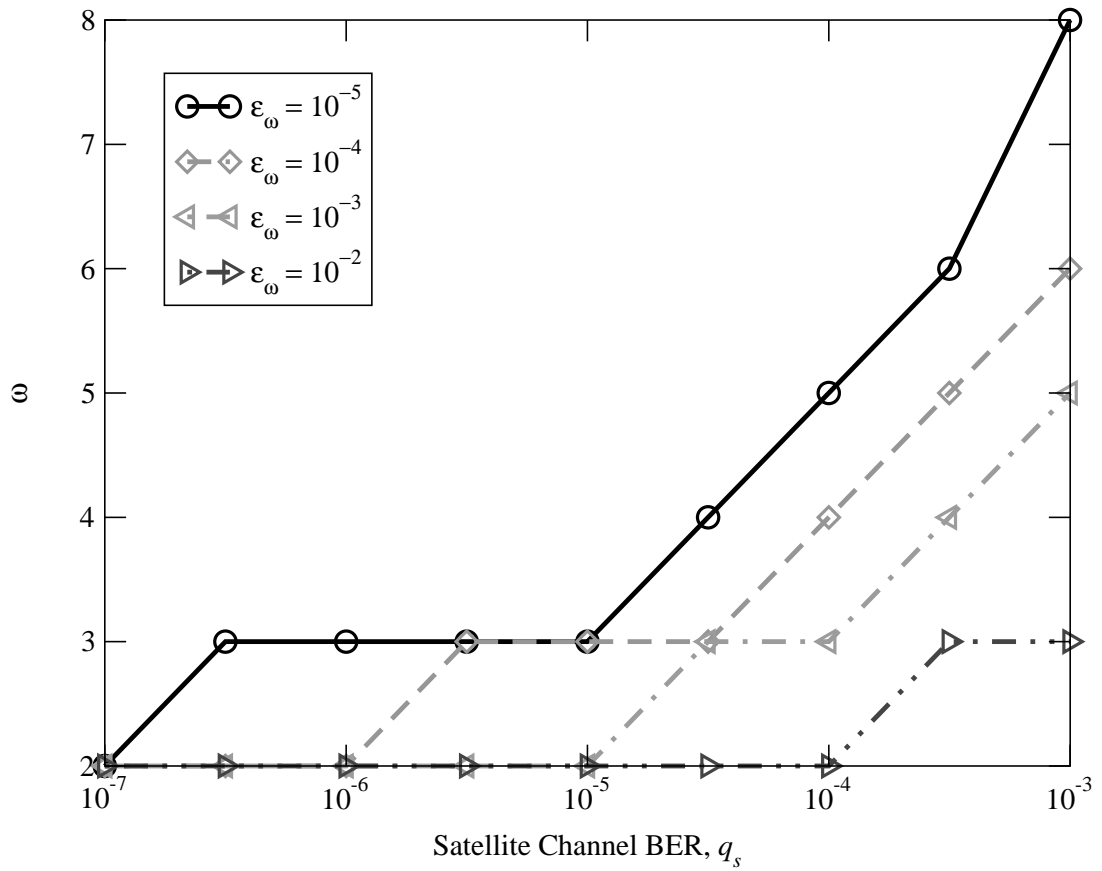


Figure 4.5: Values for ω in single-receiver, pure-satellite network (for both $\alpha_\theta = 0.25$ and $\alpha_\theta = 0.1$).

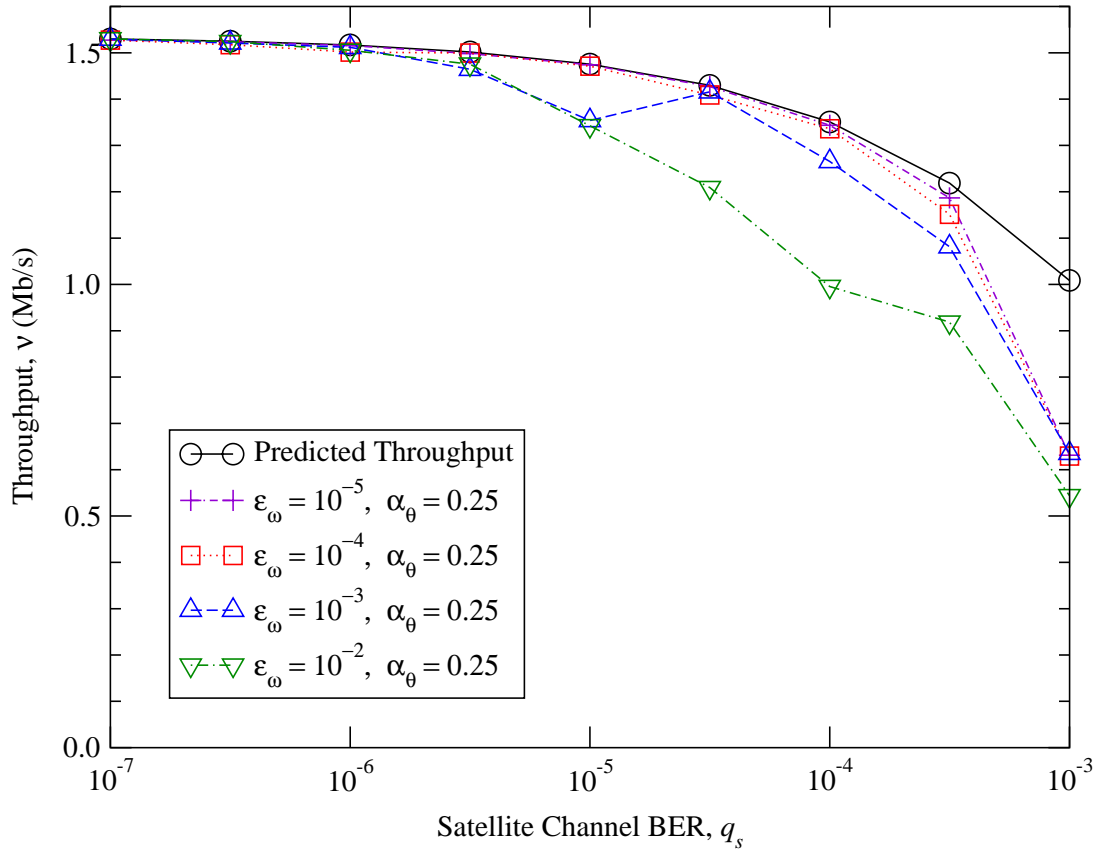


Figure 4.6: Throughput effects of ω in the single-receiver, pure-satellite network. Although omitted from the figure for clarity, throughput for $\alpha_\theta = 0.1$ is slightly less than for $\alpha_\theta = 0.25$.

over nearly the entire range of q_s of interest, it was decided to accept the shortfall at $q_s = 10^{-3}$ and to advance onward to further simulations.

4.1.3 Multiple-Receiver Pure-Satellite Network

The optimal packet length for the pure-satellite network is obtained from (3.2) and (3.8). Applying these formulas yields the results shown in Figure 4.7. Note that the optimal packet length ℓ^* decreases as q_s or M increases. This is to be expected since, in general, increasing ℓ , q_s , or M reduces throughput. So, if q_s or M increases, throughput suffers, while reducing ℓ can at least partly alleviate such throughput loss.

Of all the ARQ parameter calculations shown in Figures 3.4 and 3.5, only ω is explicitly affected by M . It is true, though, that other ARQ parameters are affected by M indirectly, since the value of ℓ is obtained by maximizing the predicted throughput expression, and the resulting ℓ determines p_s , which influences the other ARQ parameter values. Yet, experience with the single-receiver case indicates the window size, as determined by ω , has the most substantial affect upon throughput. Since $\epsilon_\omega = 10^{-5}$ was found to yield simulated throughputs fairly similar to the predicted results in the single-receiver case, it was decided to try this same ϵ_ω -value for the multiple-receivers cases.

The simulated throughputs obtained are shown in Figure 4.8. The figure indicates generally good agreement with the predicted throughput values, and that, as predicted, the throughput diminishes as the number of receivers in the network increases. Since a disparity between predicted and simulated results was observed in the single-receiver network at $q_s = 10^{-3}$, finding similar disparities for networks of a greater number of receivers, as shown in the figure, is understandable.

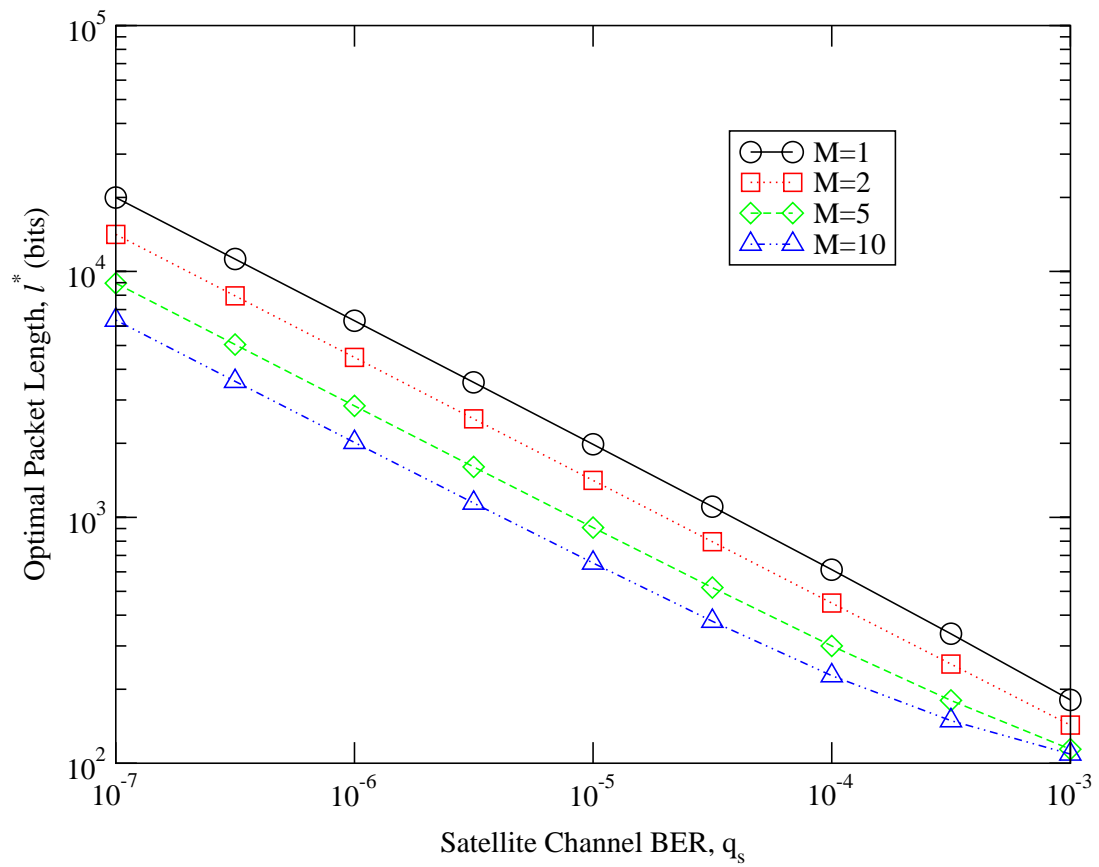


Figure 4.7: Optimal packet length ℓ^* for ARQ multicasting in a pure-satellite network ($h = 40$ bits).

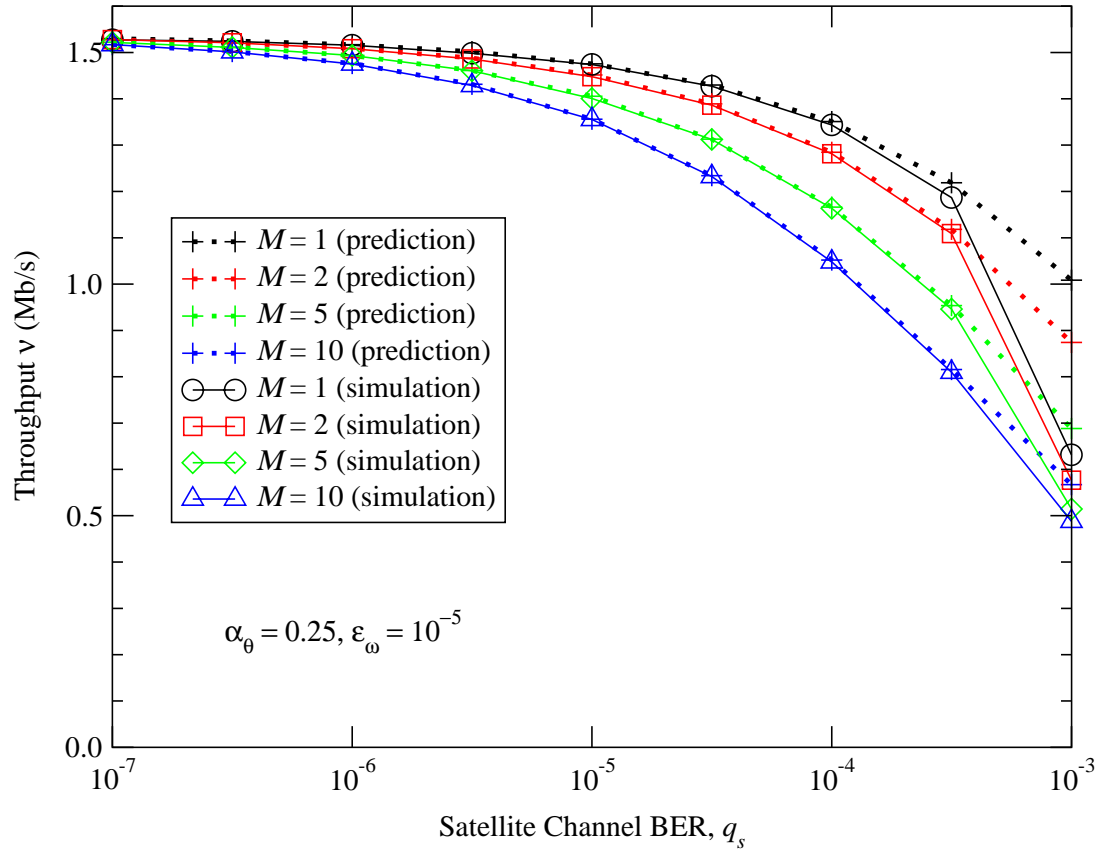


Figure 4.8: Throughputs for multiple-receiver pure-satellite networks.

4.1.4 Single-Receiver Hybrid Network

4.1.4.1 Unstable Operation Allowable

The optimal packet length for the hybrid network is obtained as prescribed in Step 1 of Figure 3.6. For the present, unstable operation will be allowed. This procedure yields results as shown in Figure 4.9.

Note from Figure 4.9 that the optimal packet length in the hybrid network is nearly the same as the single-receiver pure-satellite network's optimal packet length shown in Figure 4.1. This similarity may be explained in the following way: the packet length optimization for the pure-satellite network seeks to maximize the number of information bits delivered by transmitting a frame upon the satellite link. This is subsumed in the hybrid network by minimizing β . Hence, optimizing the packet length for the satellite link helps reduce the number of information bits which must be retransmitted terrestrially in a hybrid network. Further, only a fraction of transmitted frames must be retransmitted terrestrially, so the degree to which a packet length may be suboptimal for the terrestrial network does not have as much significance for optimizing throughput as does packet length suboptimality for the satellite link. So, optimizing the packet length for the satellite link yields a packet length which is close to the optimal packet length for the hybrid network. Of course, this description ignores the effect of a finite window, as described in the previous chapter, and which will be explored further below.

Figure 4.9 shows the optimal packet length for a given combination of q_s and q_t . For these optimal packet lengths, the corresponding FERs, p_s and p_t , are shown in Figures 4.10 and 4.11.

The predicted throughputs corresponding to the optimal packet lengths of Fig-

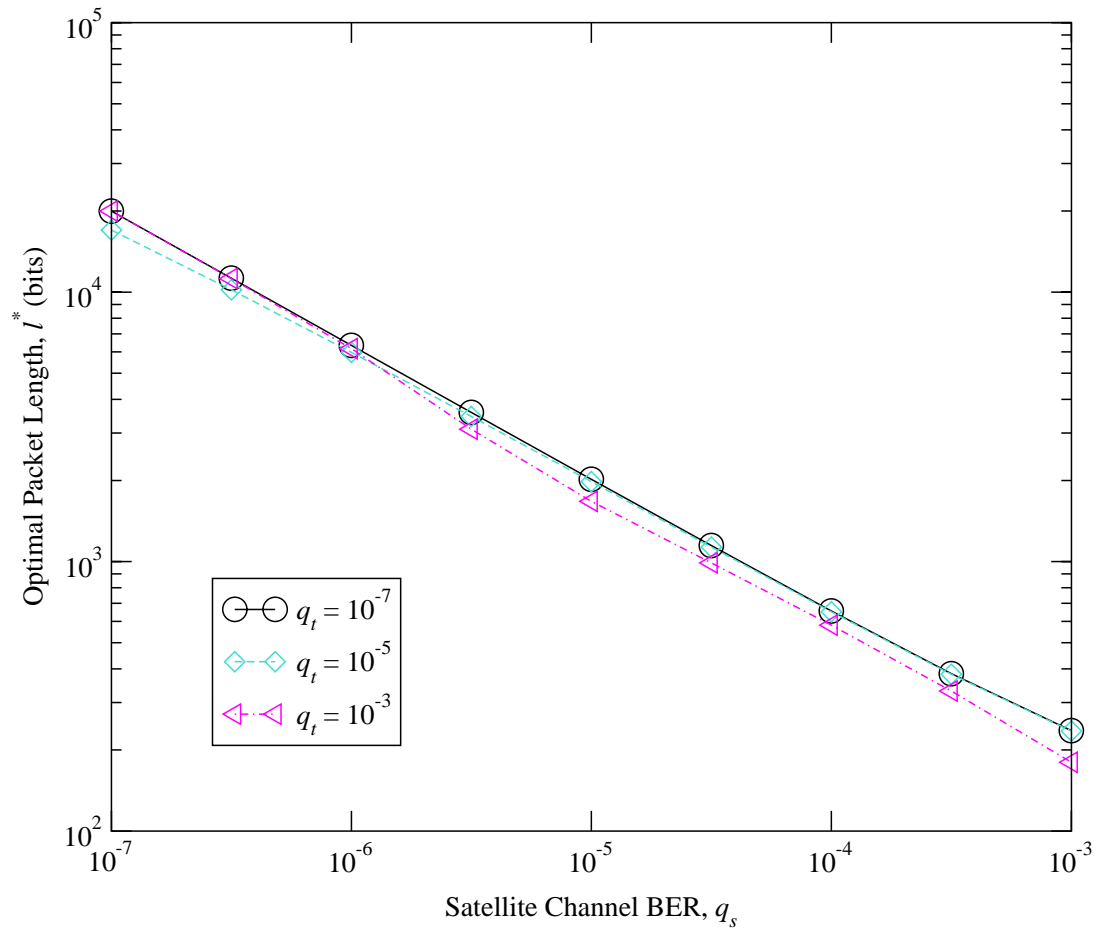


Figure 4.9: Optimal packet length for ARQ operation in the hybrid network ($h = 40$ bits).

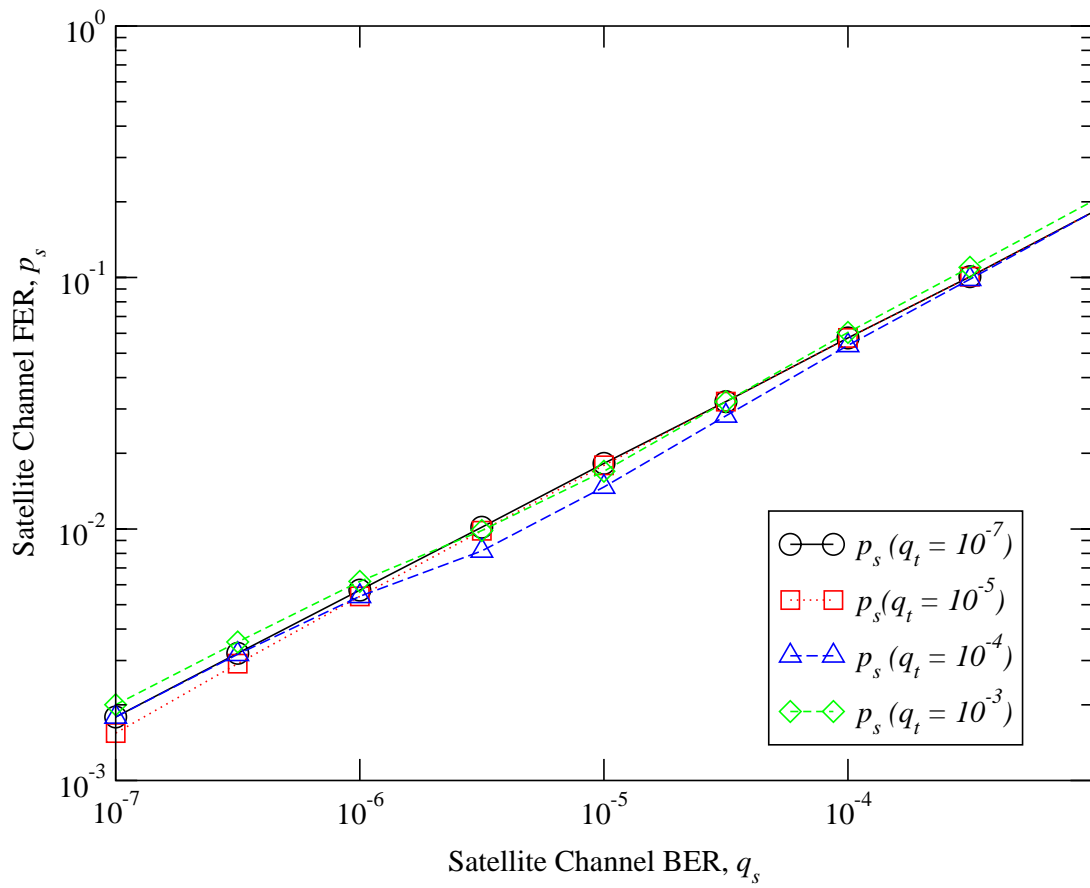


Figure 4.10: Satellite channel FER, p_s , corresponding to optimal packet lengths ($h = 40$ bits).

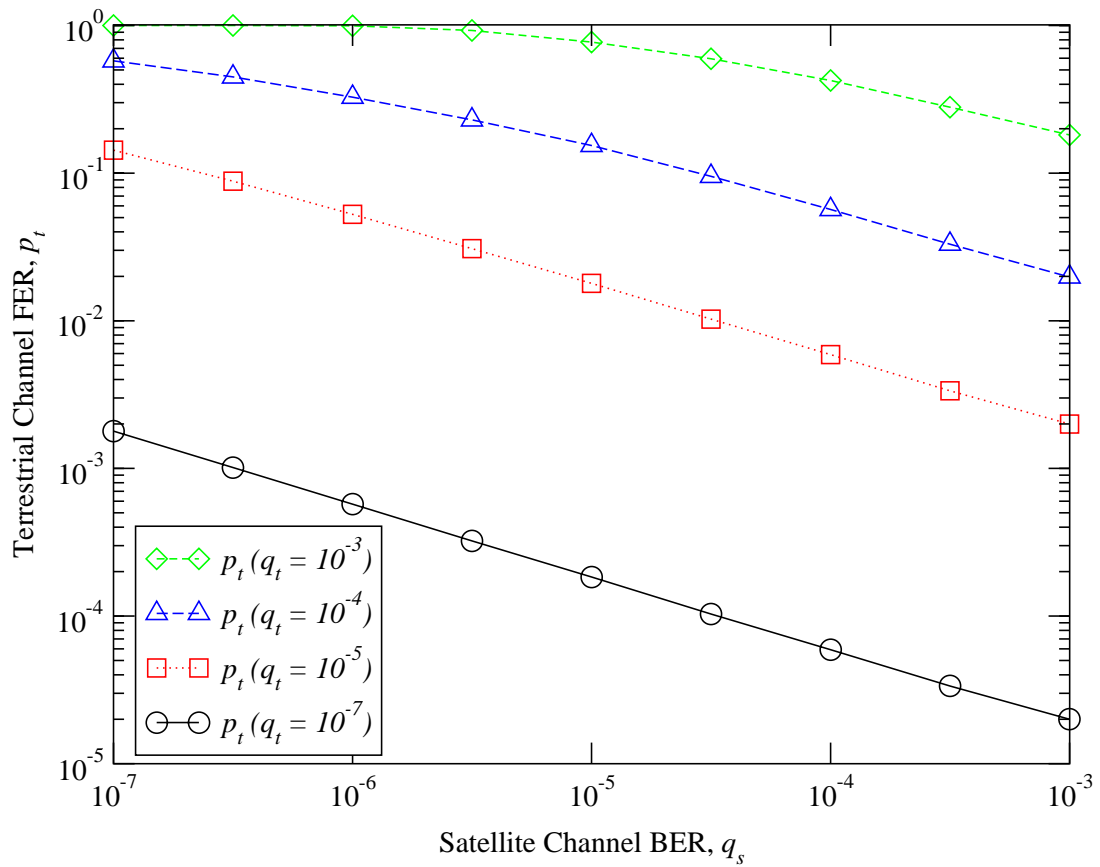


Figure 4.11: Terrestrial channel FER, p_t , corresponding to optimal packet lengths ($h = 40$ bits).

ure 4.9 are shown Figure 4.12. (A terrestrial BER of 10^{-3} is not considered in this figure since the corresponding FERs are so close to unity that window exhaustion is guaranteed if $p_s > 0$. This observation indeed conflicts with (2.7) but that expression was developed without regard to a finite window size.) The throughput curves are almost exactly coincident for all values of q_t considered, indicating little predicted sensitivity of throughput to the terrestrial channel bit error rate. These throughput predictions are idealized, though, for they assume unlimited window sizes and no timer expirations. Such assumptions are not found in actual operation, so some sensitivity to q_t would be expected in practice.

Having obtained the optimal packet lengths for each combination of q_s and q_t , the ARQ parameters can be considered. The results from the single-receiver, pure-satellite network discussion above were used as a guide for this hybrid example. As before, the example proceeds with the ℓ -values obtained above rounded to their nearest positive multiples of eight to suit simulation constraints.

For the hybrid network, assuming $q_t = 10^{-5}$, following the calculation prescribed in Figure 3.7 yields the θ -results shown in Figure 4.13. The θ -values obtained for the hybrid network are considerably smaller than those obtained for the pure-satellite network (as shown in Figure 4.3) due to the round-trip time being smaller in the hybrid network, in which acknowledgements are sent terrestrially. The throughput effects of different values of θ are shown in Figure 4.14. In this figure, little throughput sensitivity to θ is seen. For $q_s > 10^{-5}$, the throughput curves drop sharply, and become convex. The unusual behavior exhibited for $q_s > 10^{-5}$ corresponds to unstable operation, which, as described in the previous chapter, results in the satellite transmitter idling at times, waiting for the terrestrial transmitter to successfully deliver the earliest frame in the window. That discussion also asserted that increasing the window size

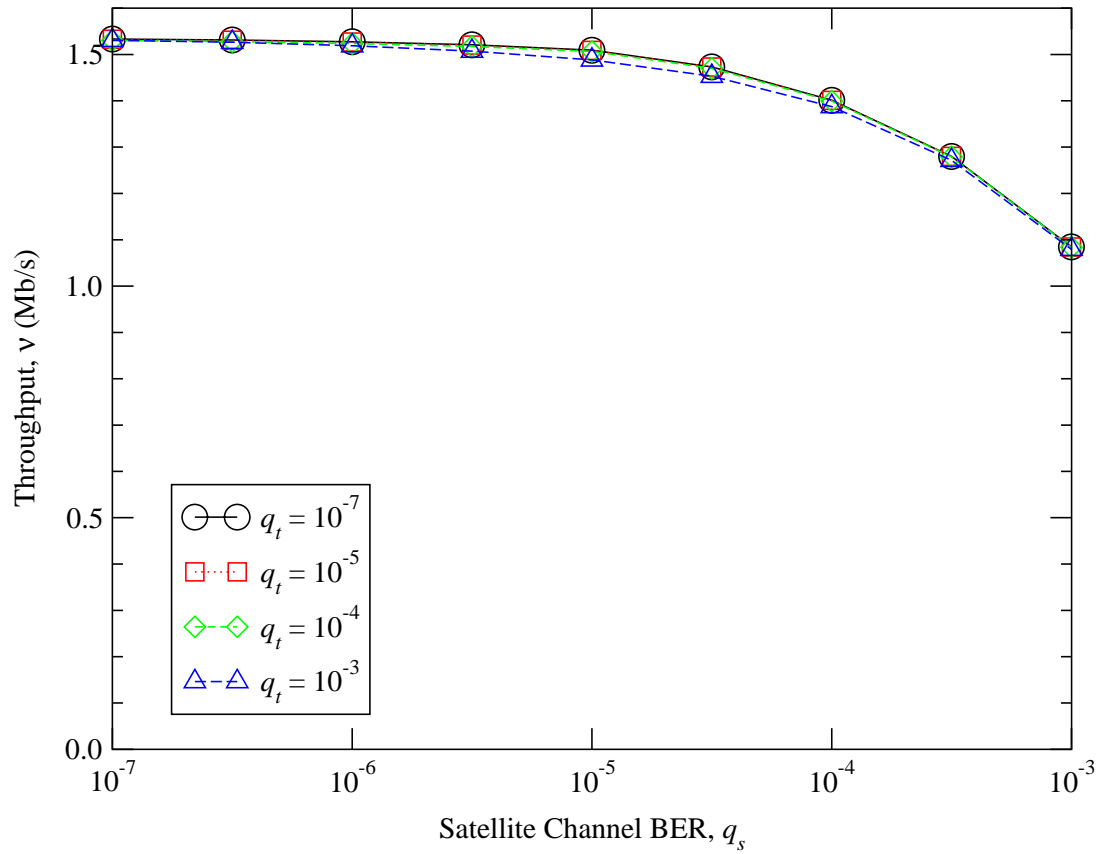


Figure 4.12: Predicted throughput for ARQ operation in the hybrid network, using optimal packet lengths from Figure 4.9.

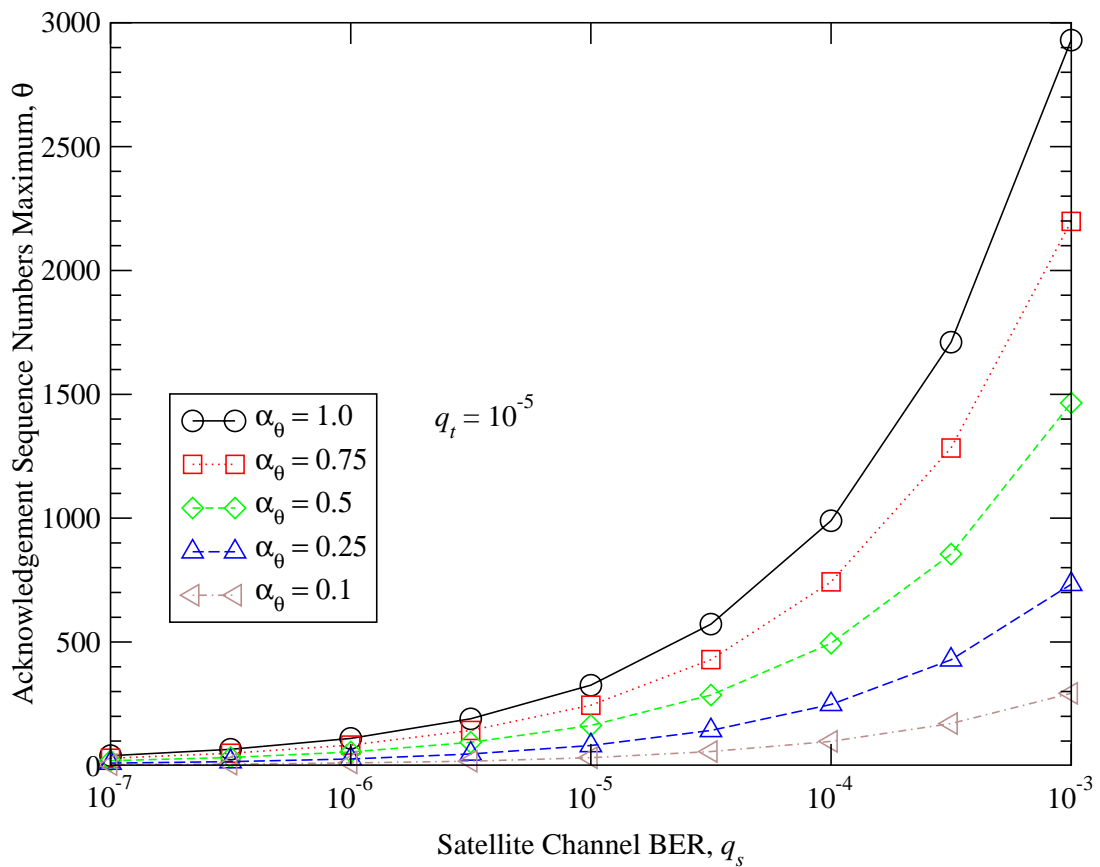


Figure 4.13: Acknowledgement sequence numbers maximum, θ , for the hybrid network.

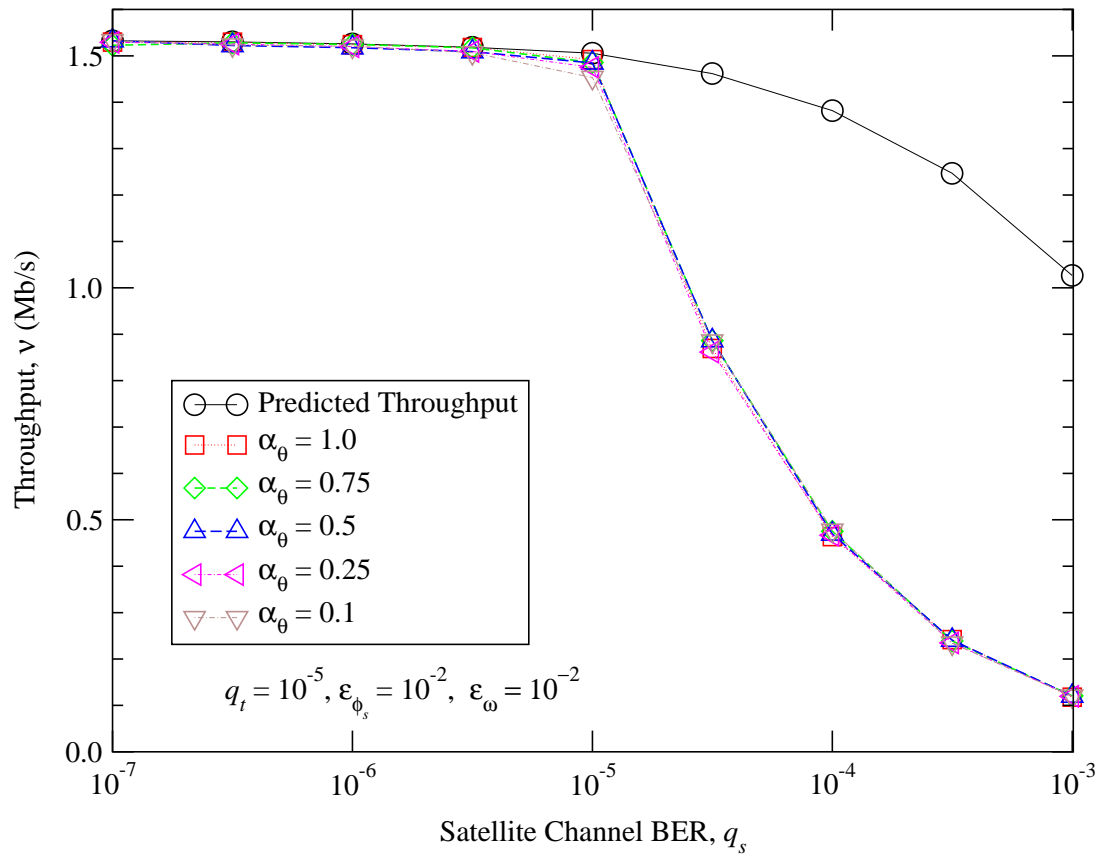


Figure 4.14: Throughput effects of θ in the hybrid network.

does not improve matters, and simulation results to be presented below will verify this assertion.

In the figure, at $q_s = 10^{-5}$, insignificant throughput benefit was observed by varying α_θ . Simulations conducted at lesser and greater values of q_t yielded similar results. These results suggest that resources can be conserved by setting $\alpha_\theta = 0.1$. However, to parallel pure-satellite network simulations presented above, hereafter α_θ was set to 0.25.

After obtaining θ , ϕ_s and ϕ_t are to be obtained. As in the pure-satellite network, varying ϵ_{ϕ_s} and ϵ_{ϕ_t} from 10^{-1} to 10^{-12} did not affect the values of ϕ_s and ϕ_t obtained. The two parameters of ϕ_s and ϕ_t were accordingly set hereafter to $\lceil r_s K_{ack}^{max} / r_a L \rceil$ and $\lceil r_t K_{ack}^{max} / r_a L \rceil$, respectively, for hybrid network simulations.

Next, the effect of different window sizes was considered. For $q_t = 10^{-5}$, the values of ω obtained are shown in Figure 4.15. The dip in the curve for $\epsilon_\omega = 10^{-5}$ indicates the transition between which of the first two quantities in (3.24) yields the maximum.

The corresponding simulation results are shown in Figure 4.16. This figure indicates that, as asserted earlier, increasing the window size does not improve throughput. In stable operation, window exhaustion is rare, so a larger window does not help. Window exhaustion is common in unstable operation, but once the satellite transmitter idles, the duty cycle of that transmitter depends on the transmission and error rates of the two links, but not on the window size. So, if the operation of the hybrid network is unstable, extending the window does not improve the throughput.

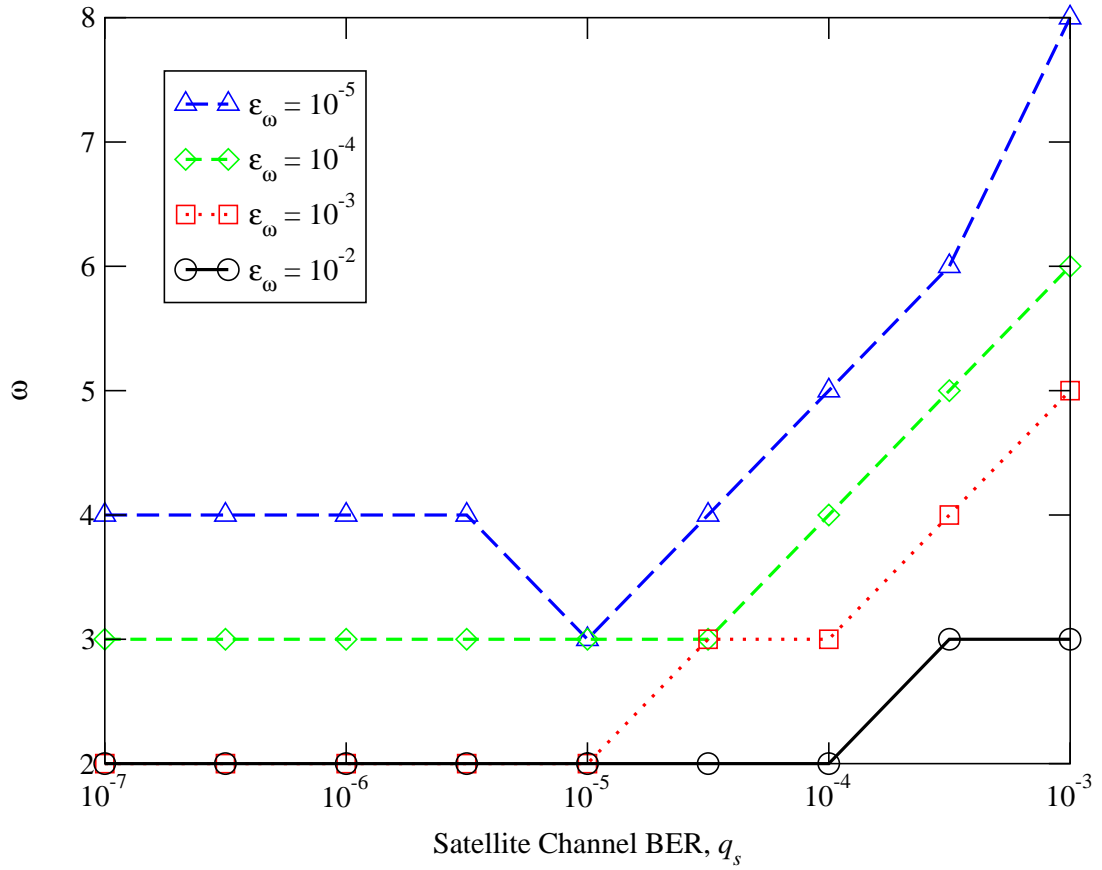


Figure 4.15: Values of ω for the hybrid network ($q_t = 10^{-5}$).

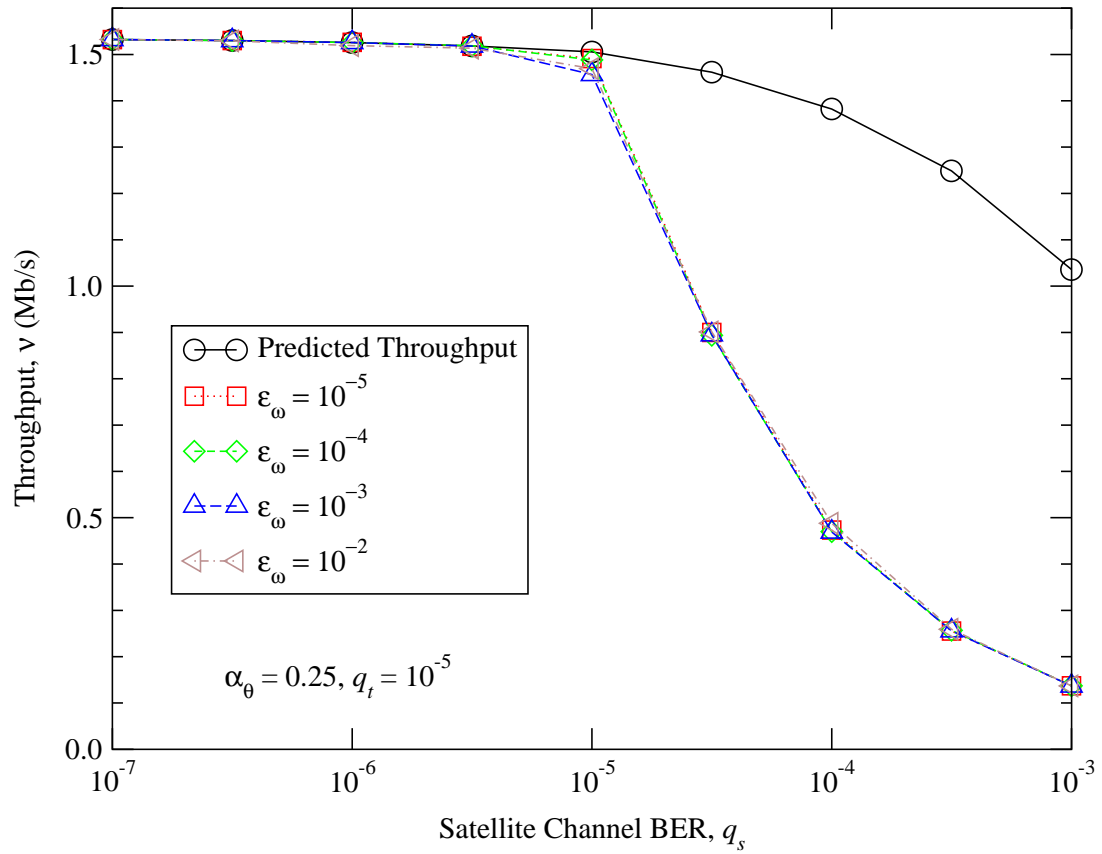


Figure 4.16: Throughput effects of ω in the hybrid network ($q_t = 10^{-5}$).

4.1.4.2 Stability–Assured Operation

In an effort to improve throughput for $q_s > 10^{-5}$, the hybrid network was examined with ℓ calculated to assure stability. Recalculating the packet lengths yields the values shown in Figure 4.17. The optimal packet lengths calculated earlier, and shown in Figure 4.9, are shown as well for comparison. For $q_t \leq 10^{-5}$ and $q_s \leq 10^{-5}$, the packet lengths calculated above earlier yield stable operation. As q_s increases beyond 10^{-5} , the packet lengths calculated above do not yield stable operation, and the disparity between the earlier curves and the “stable-only” curves grows. For $q_t = 10^{-3}$, none of the values calculated earlier yield stable operation, and so the “stable-only” curve does not anywhere coincide with its earlier counterpart. This indicates that for $q_t = 10^{-3}$, the theoretically optimal packet length cannot provide stable operation, and so the procedure for calculating ℓ from (3.4) to yield stable operation must be employed.

In this figure, as in those which follow, the curves for quantities related to stability-assured operation do not extend to $q_s = 10^{-3}$ since stable operation is unachievable for so high a value of q_s .

The FERs p_s and p_t corresponding to the stability-assured packet lengths shown in Figure 4.17 are given in Figures 4.18 and 4.19, respectively. The transition between “ordinary” and “stability-assured” operation is clearly visible in the first of these two figures.

The predicted throughputs for the calculated packet lengths are shown in Figure 4.20. Note that Figure 4.19 indicates $p_t \approx 1$ for $q_t = 10^{-3}$ if q_s is less than 10^{-6} or so. As explained above, if p_t is nearly unity, then window exhaustion will eventually result if $p_s > 0$, and so throughput will be nearly zero for such cases. With this observation, and noting that Figure 4.20 indicates identical throughputs for $q_t = 10^{-7}$ and $q_t = 10^{-5}$, subsequent discussion concentrates on cases of $q_t = 10^{-5}$.

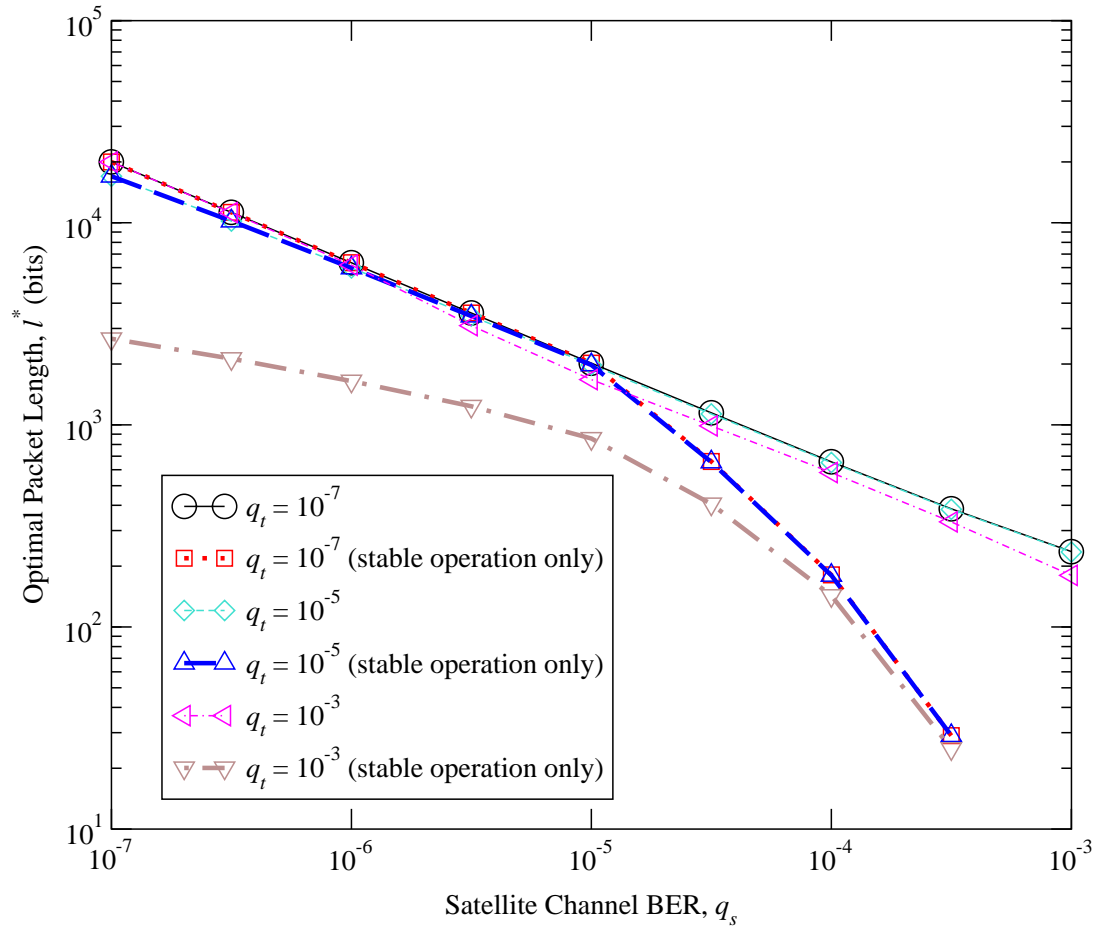


Figure 4.17: Optimal packet length, l , for stable operation in the hybrid network ($h = 40$ bits).

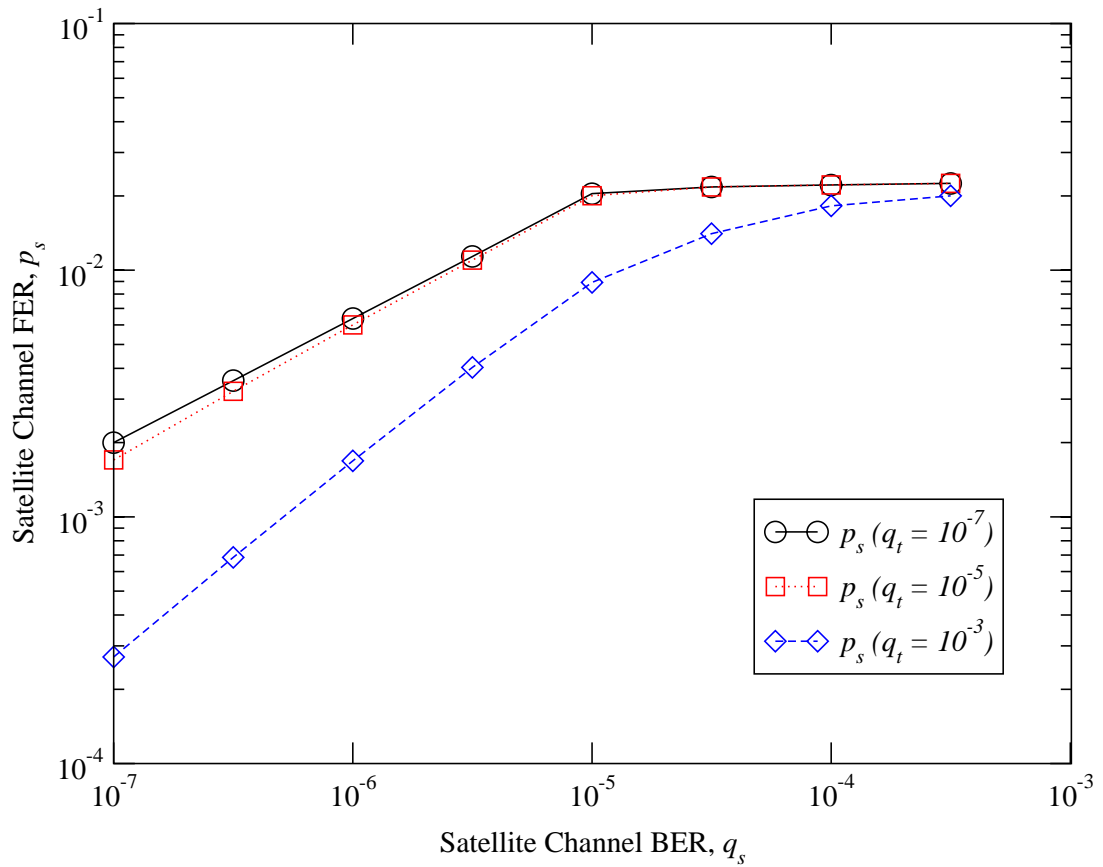


Figure 4.18: Satellite channel FER, p_s , corresponding to the optimal packet lengths calculated to assure stable operation in the hybrid network ($h = 40$ bits).

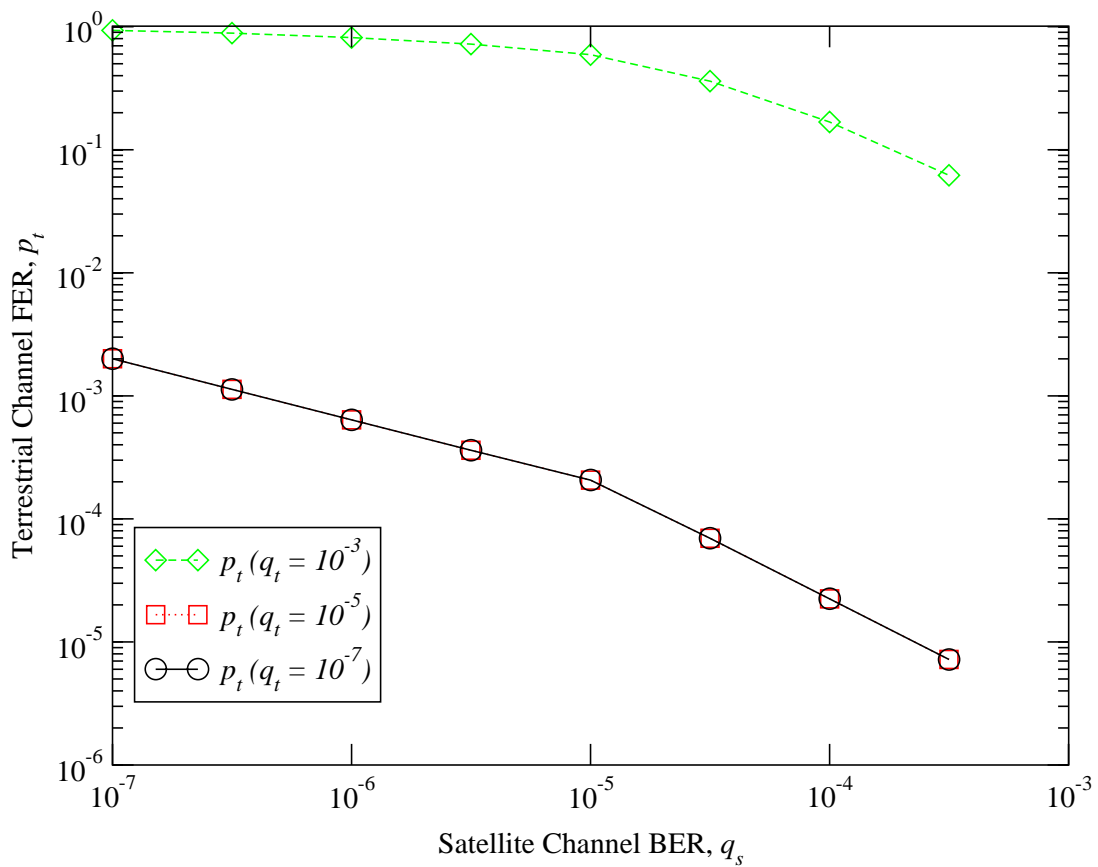


Figure 4.19: Terrestrial channel FER, p_t , corresponding to the optimal packet lengths calculated to assure stable operation in the hybrid network ($h = 40$ bits).

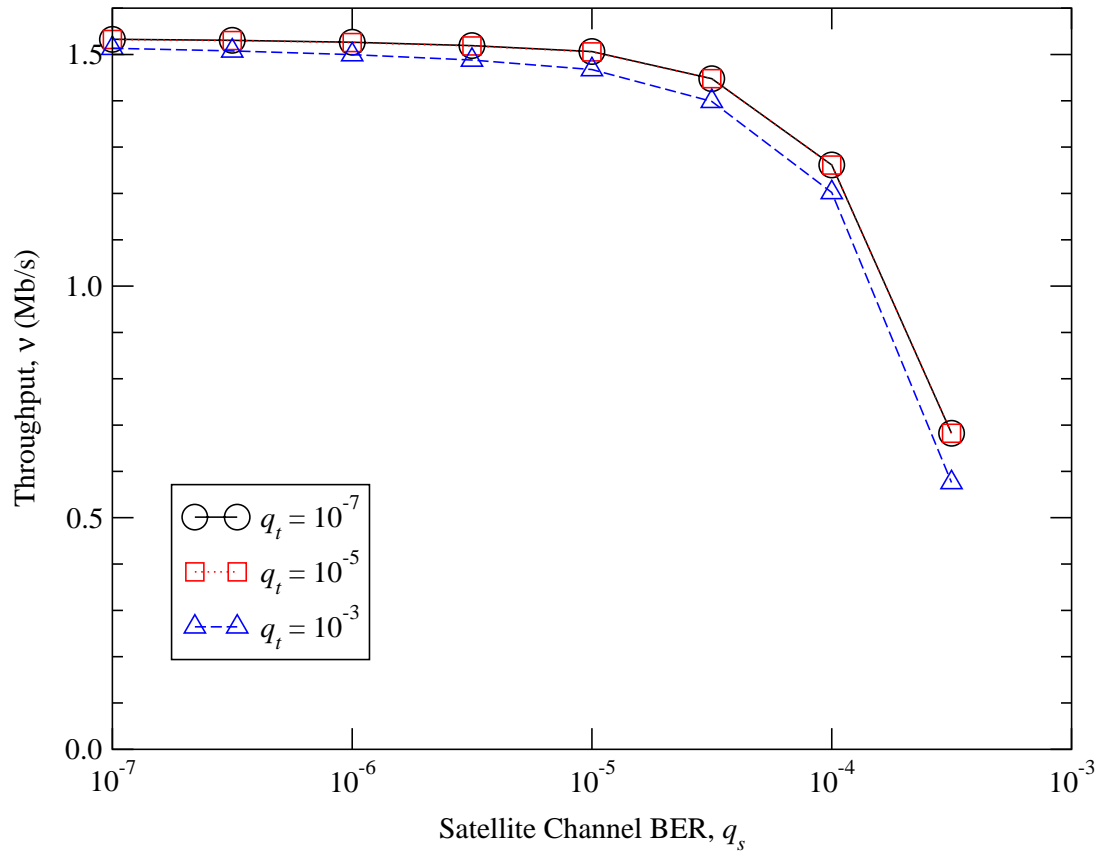


Figure 4.20: Predicted throughput for ARQ operation in the hybrid network, using optimal packet lengths from Figure 4.17.

With ℓ recalculated to assure stable operation, the values of θ obtained for a range of α_θ are shown in Figure 4.21. The values of θ shown in this figure are larger for $q_s \geq 10^{-5}$ than before since now smaller packet lengths were calculated to assure stability for these higher satellite channel BERs. The corresponding effect of θ upon throughput for $q_t = 10^{-5}$ is shown in Figure 4.22. Although some variation in throughput with α_θ is visible in Figure 4.22, this variation is slight. Hence, and to parallel earlier simulations, hereafter α_θ was set to 0.25.

As before, varying ϵ_ϕ over a wide range yielded no change in the values of ϕ_s and ϕ_t calculated.

Calculating ℓ to assure stability yielded the values of ω shown in Figure 4.23. The values obtained for $q_s > 10^{-5}$ are less than those obtained earlier in which unstable operation was accepted. The reason smaller values were obtained for ω in the present situation is that calculating ℓ to assure stability yields smaller values for ℓ , and so smaller values for p_s and p_t , which in turn yield smaller values for ω by (3.24).

The simulation results corresponding to the different values of ω are shown in Figure 4.24. As in the pure-satellite case, the figure indicates enlarging the window improves performance, and excellent agreement of simulated throughput with the predicted performance can be achieved with sufficiently large window sizes.

4.1.5 Multiple-Receiver Hybrid Network

The optimal packet lengths for the multicast hybrid network are the same as for the single-receiver hybrid network, which were given above in Figures 4.9 and 4.17. Using the packet lengths shown in Figure 4.17, namely the packet lengths calculated to assure stability, multi-receiver hybrid network simulations were conducted. These simulations used $q_t = 10^{-5}$, $\alpha_\theta = 0.25$, and $\epsilon_\omega = 10^{-5}$, which are the same param-

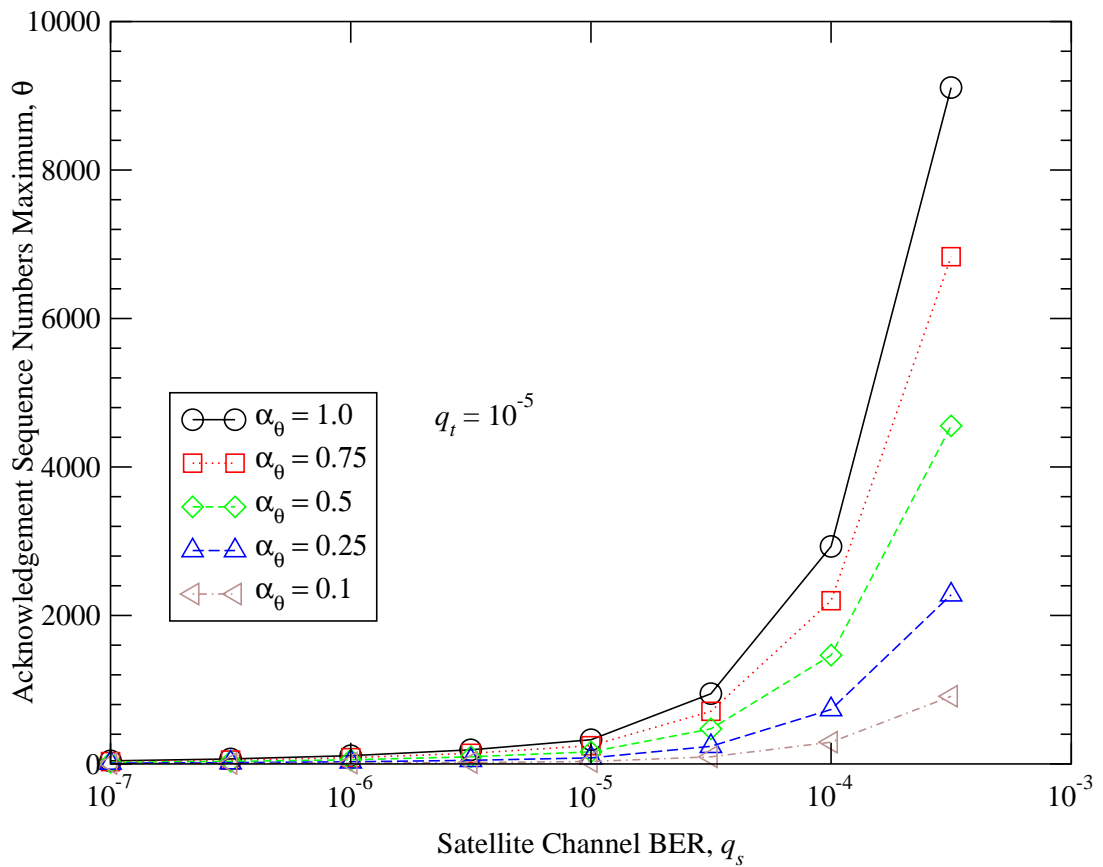


Figure 4.21: Acknowledgement sequence numbers maximum, θ , for the hybrid network, with ℓ calculated to assure stable operation.

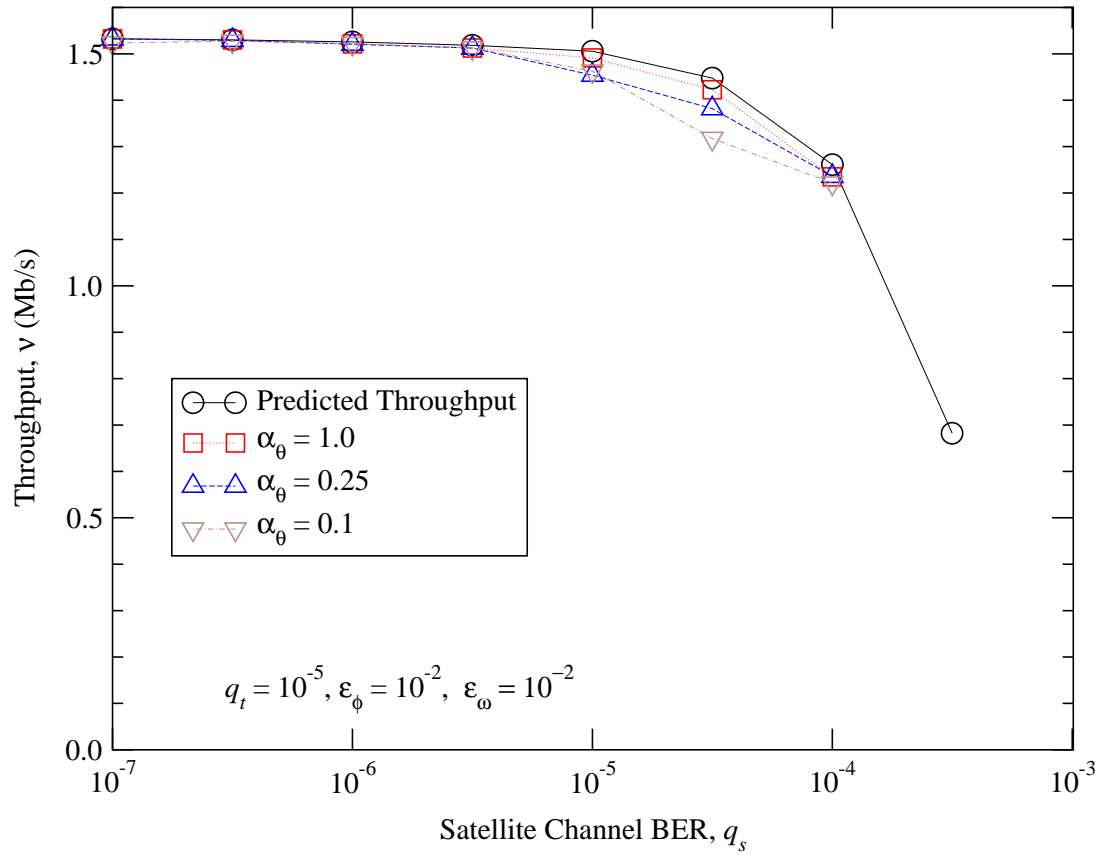


Figure 4.22: Throughput effects of θ in the hybrid network with ℓ calculated to assure stable operation.

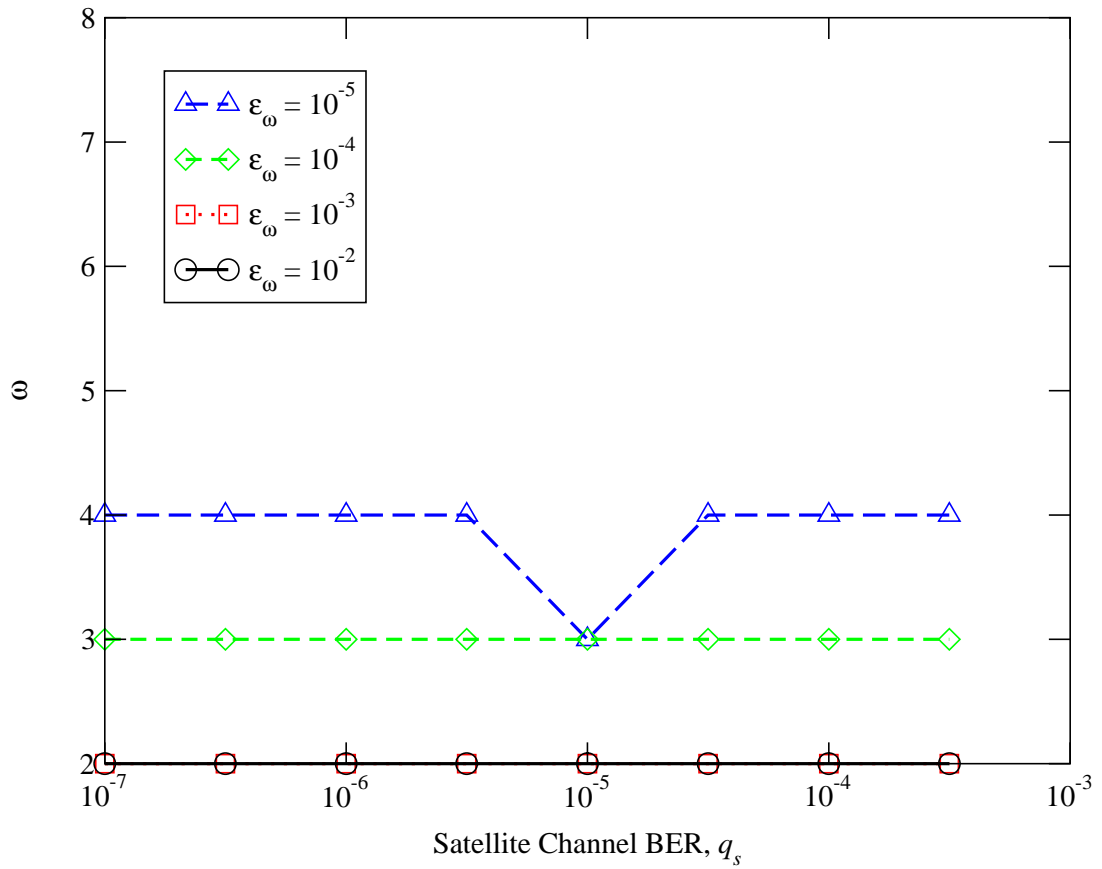


Figure 4.23: Values for ω in the hybrid network ($q_t = 10^{-5}$) with ℓ calculated to assure stable operation.

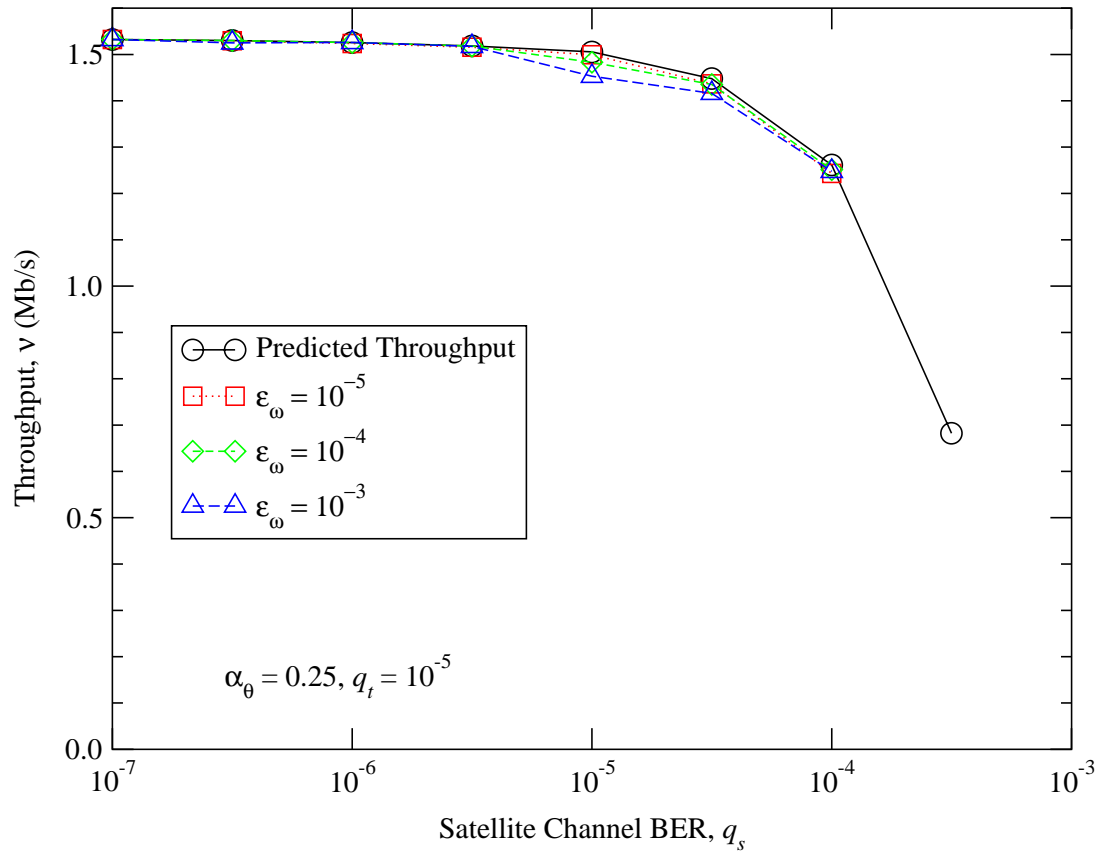


Figure 4.24: Throughput effects of ω in the hybrid network ($q_t = 10^{-5}$) with ℓ calculated to assure stable operation.

ters used for the results shown in Figure 4.24. The multi-receiver simulation results for the hybrid network are shown in Figure 4.25. The figure shows little sensitivity of throughput to the number of receivers in the hybrid network, although only limited simulation results are presented in the figure. Yet, discussion in section 3.3.3.4 suggested some sensitivity would be shown. However, such sensitivity may be partially masked by preceding efforts in which ARQ parameters were adjusted to maximize throughput. It may be expected that greater throughput sensitivity to M would be exhibited with a greater number of receivers, or at higher satellite channel BERs than for which simulation results are provided in the figure.

The reason for presenting only limited simulation results is itself an instructive matter. In general, the greater the satellite channel BER, the larger the window size and the timer period calculated by the formulas in Figures 3.4, 3.5, 3.6 and 3.7. In the hybrid network in particular, r_t is smaller than r_s (by a factor of almost 46, in these examples), and so the window size and timer period are further increased. Further, at higher satellite channel BERs, smaller packet sizes are calculated to assure stability. These observations translate to substantially more memory being required for simulating the hybrid network with a poor satellite channel. Hence results could not be obtained for the most memory-intensive simulation scenarios, despite using the best RAM-equipped computer available to this author (a Sun workstation with 1 GB of memory). While the NS simulator was found to have some constraints which limited using memory as efficiently as possible, the amount of memory required is a legitimate concern for practical implementation.

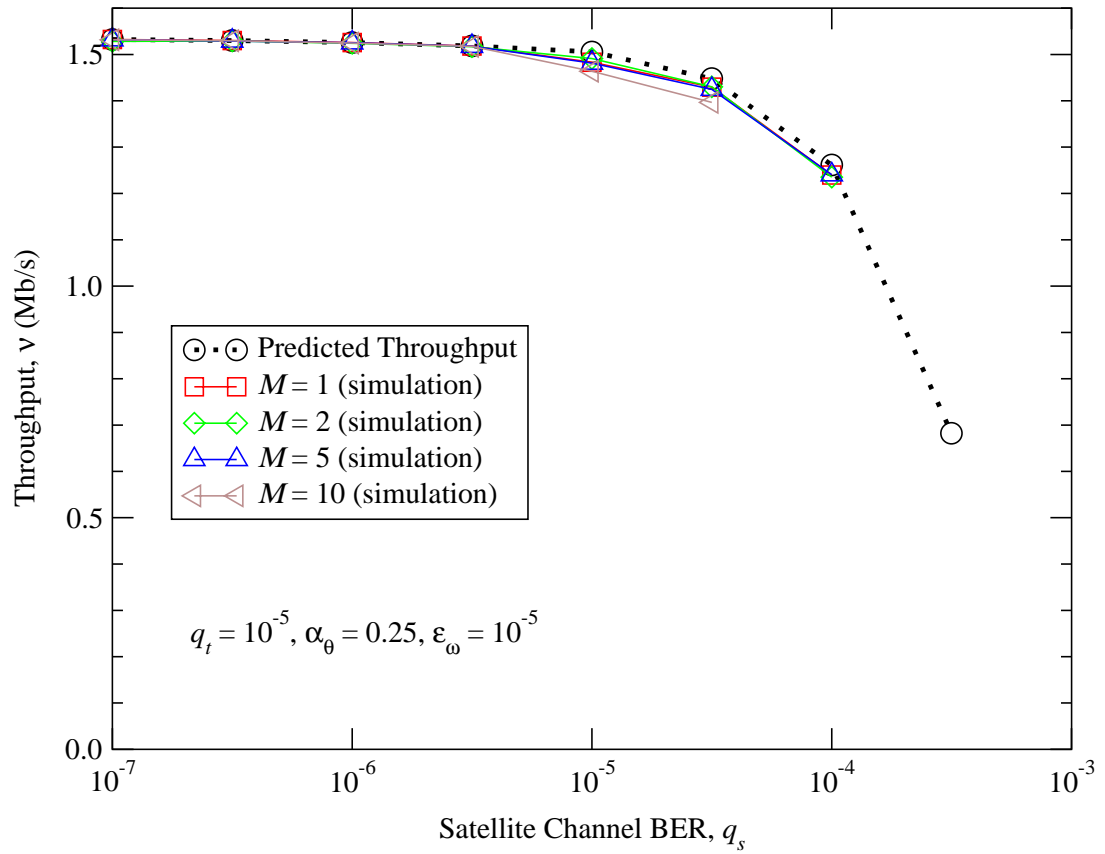


Figure 4.25: Throughputs for multiple-receiver hybrid networks.

4.1.6 Comparison of Pure-Satellite and Hybrid Networks for Multicasting to Identical Receivers

The simulation results presented above for both pure-satellite and hybrid networks indicate predicted throughputs can be achieved, or nearly achieved, if the protocol parameters are adjusted properly. Figure 4.26 compares the predicted throughputs for the two networks. The figure indicates ARQ multicast throughput can be improved by supplementing a pure-satellite network with terrestrial links to form a hybrid network, and modifying the multicast ARQ protocol to take appropriate advantage of these links. At very high satellite link BERs, though, the hybrid network cannot support stable operation. Also, if the satellite link BER is low enough to allow stable hybrid network operation, but is still relatively high, the hybrid network may yield poorer throughput than the pure-satellite network, since many retransmissions for a receiver in the hybrid network must be conducted over a terrestrial link slower than the satellite link. Yet, for lesser satellite link BERs, the hybrid network offers superior throughput.

The throughput advantage of the hybrid network of course requires additional bandwidth. In Chapter 2, the throughput efficiency, η , was defined as the throughput (v) divided by the sum of the network link bandwidths used to achieve that throughput. The throughput efficiencies corresponding to the pure-satellite and hybrid network throughputs of Figure 4.26 are given in Figure 4.27. (The throughput curves in Figure 4.26 are very similar for different terrestrial BERs in the hybrid network; a BER of $q_t = 10^{-4}$ was deemed representative for composing Figure 4.27.) This figure indicates the pure-satellite network is more efficient than the hybrid network. This conclusion might appear to contradict Figure 2.6's implication of a region of satellite link BERs in which the hybrid network is more efficient. However, Figure 2.6 assumes a fixed packet length, while Figure 4.27 reflects using an optimal packet length

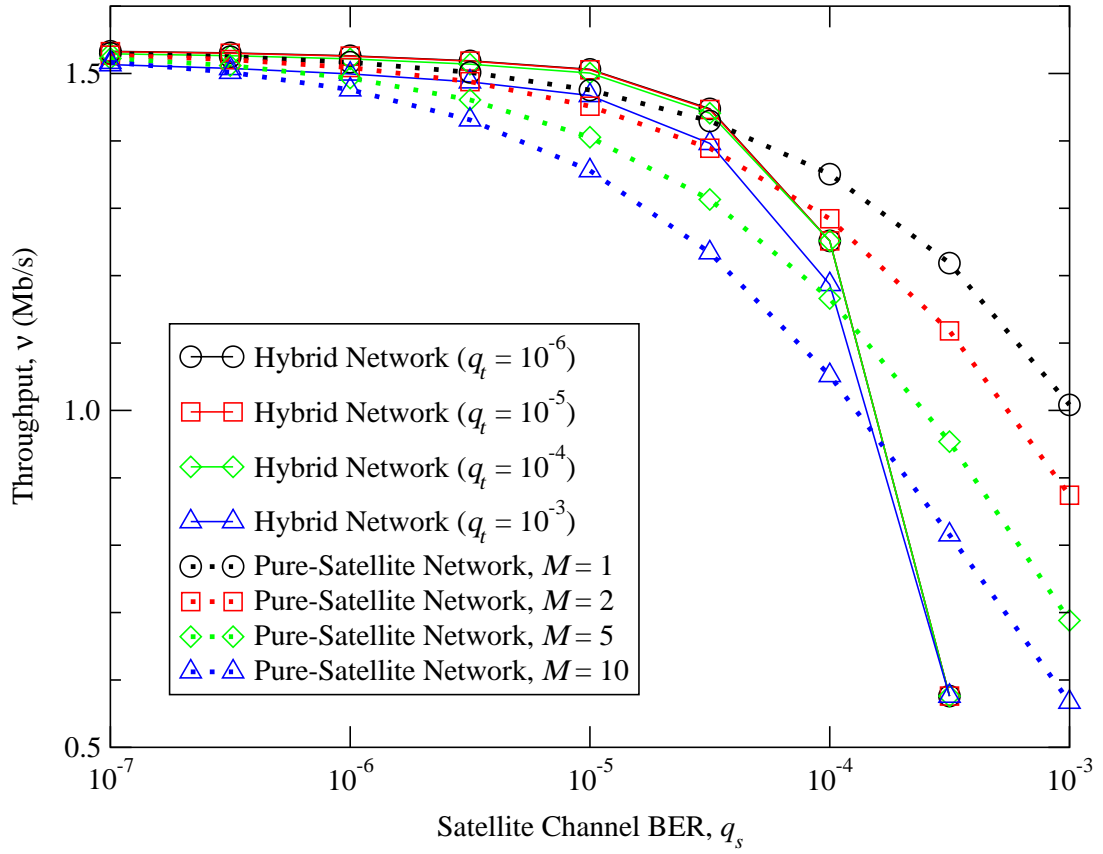


Figure 4.26: Throughput comparison for the pure-satellite and hybrid networks.

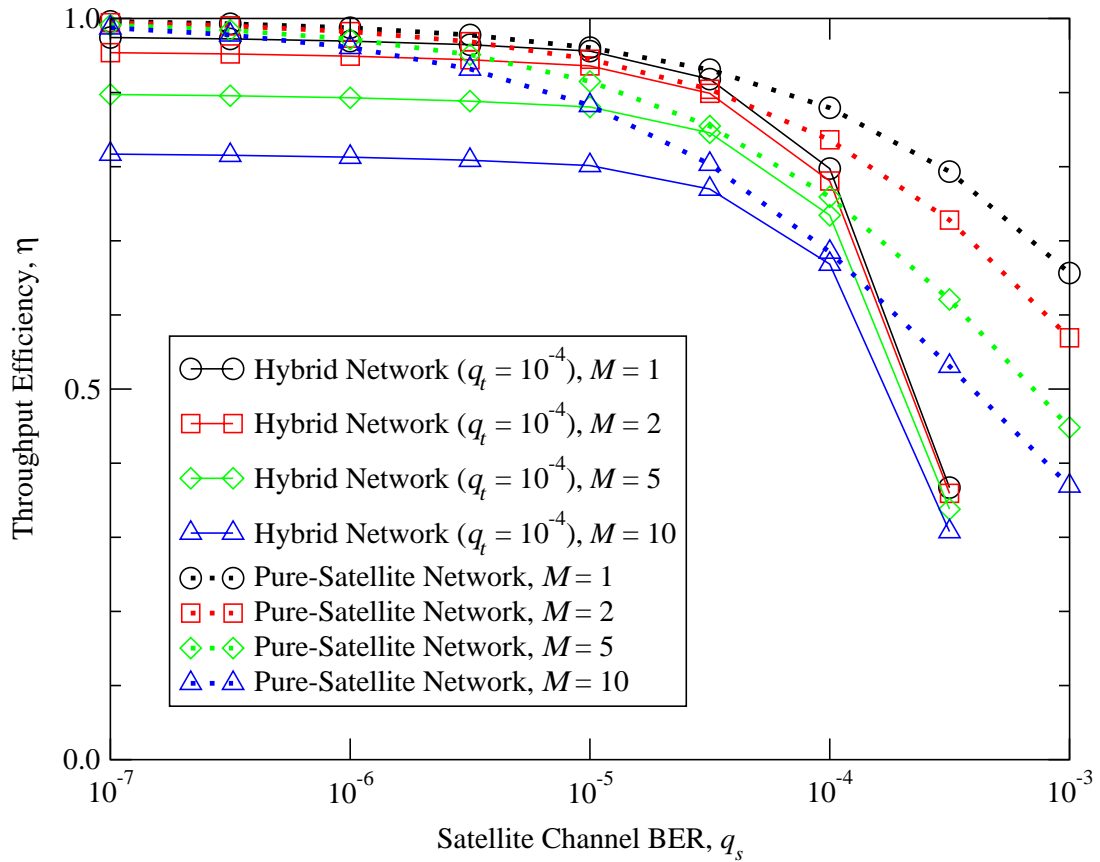


Figure 4.27: Throughput efficiency comparison for the pure-satellite and hybrid networks.

for each satellite link BER, or combination of satellite and terrestrial link BERs, for the hybrid network. Still, Figure 4.27 suggests the terrestrial channels of the hybrid network could possibly be used more efficiently.

4.2 Bandwidth Splitting

4.2.1 Problem Formulation

Having shown that a hybrid network can in some cases offer better throughput for multicasting than can a pure-satellite network, it is natural to wonder about possible optimizations to improve throughput. Beyond the discussions of the previous chapter, another matter of interest is optimizing the amounts of bandwidth used in the satellite and terrestrial links, which is now addressed.

Let r_{tot} denote a given total amount of bandwidth to be devoted for carrying information frames from a transmitter to M receivers. This bandwidth is to be partitioned into an amount for the satellite link, r_s , and the balance is to be assigned to M terrestrial links, each having bandwidth r_t . Let ρ , $0 \leq \rho \leq 1$, be a splitting factor, defined as the fraction of total bandwidth assigned to terrestrial links. With M receivers, each receiver is to be assigned a terrestrial bandwidth of $r_t = \rho r_{tot}/M$. The question of present interest is to find the value of ρ which maximizes throughput.

A slightly more interesting question, which does not affect the outcome of the one just posited, is to apply a cost constraint. A network designer may have D dollars available for links, where the cost per unit bandwidth is C_s for a satellite link and C_t for a terrestrial link. Hence the constraint is $C_s(1 - \rho)r_{tot} + C_t M \rho r_{tot}/M \leq D$, which yields $r_{tot} \leq D/(C_s(1 - \rho) + C_t \rho)$. So, once ρ is found, r_{tot} can be determined, in turn yielding the appropriate satellite link bandwidth, $r_s = (1 - \rho)r_{tot}$, and terrestrial link

bandwidth per receiver, $r_t = \rho r_{tot}/M$.

To address the original question of finding the value of ρ which maximizes throughput, assume a fixed amount of overhead, h , per information frame. Also assume that the link BERs q_s and q_t are given. These quantities are assumed to be fixed, independent of the link transmission speed. To avoid trivial cases, the link BERs are assumed to satisfy $0 < q_s, q_t < 1$.

4.2.2 Relation of Splitting Factor to Throughput

4.2.2.1 Fixed Packet Length

To understand the effect of ρ upon throughput, it will be helpful to temporarily assume a fixed packet length ℓ . The throughput then varies with ρ as sketched in Figure 4.28, as will now be explained.

If $\rho = 0$, a pure satellite network is described; $\rho = 1$ specifies a purely terrestrial network; and $0 < \rho < 1$ describes a hybrid network. Since the three cases of $\rho = 0$, $0 < \rho < 1$, and $\rho = 1$ correspond to different networks, the throughput as a function of ρ is discontinuous at $\rho = 0$ and $\rho = 1$.

In the pure-satellite network, corresponding to $\rho = 0$, the throughput is $\ell r_{tot}/(\ell + h)\beta_{sat,M}$. As shown in section 2.2.1 (page 25), $\beta_{sat,M}$ declines with M because more retransmissions are required for more receivers. Hence the throughput for $\rho = 0$ may exceed that achievable in the hybrid network if M is small, but declines as M increases.

If $\rho = 1$, then no satellite link is employed, and there is no multicasting. Hence this case corresponds to a star-topology terrestrial network of M links connecting the transmitter to each receiver, and each link has bandwidth r_{tot}/M . This is effectively an M -way unicast network. As seen by each receiver, this situation is similar to that in a single-receiver pure-satellite network, with a notable difference being the link BER

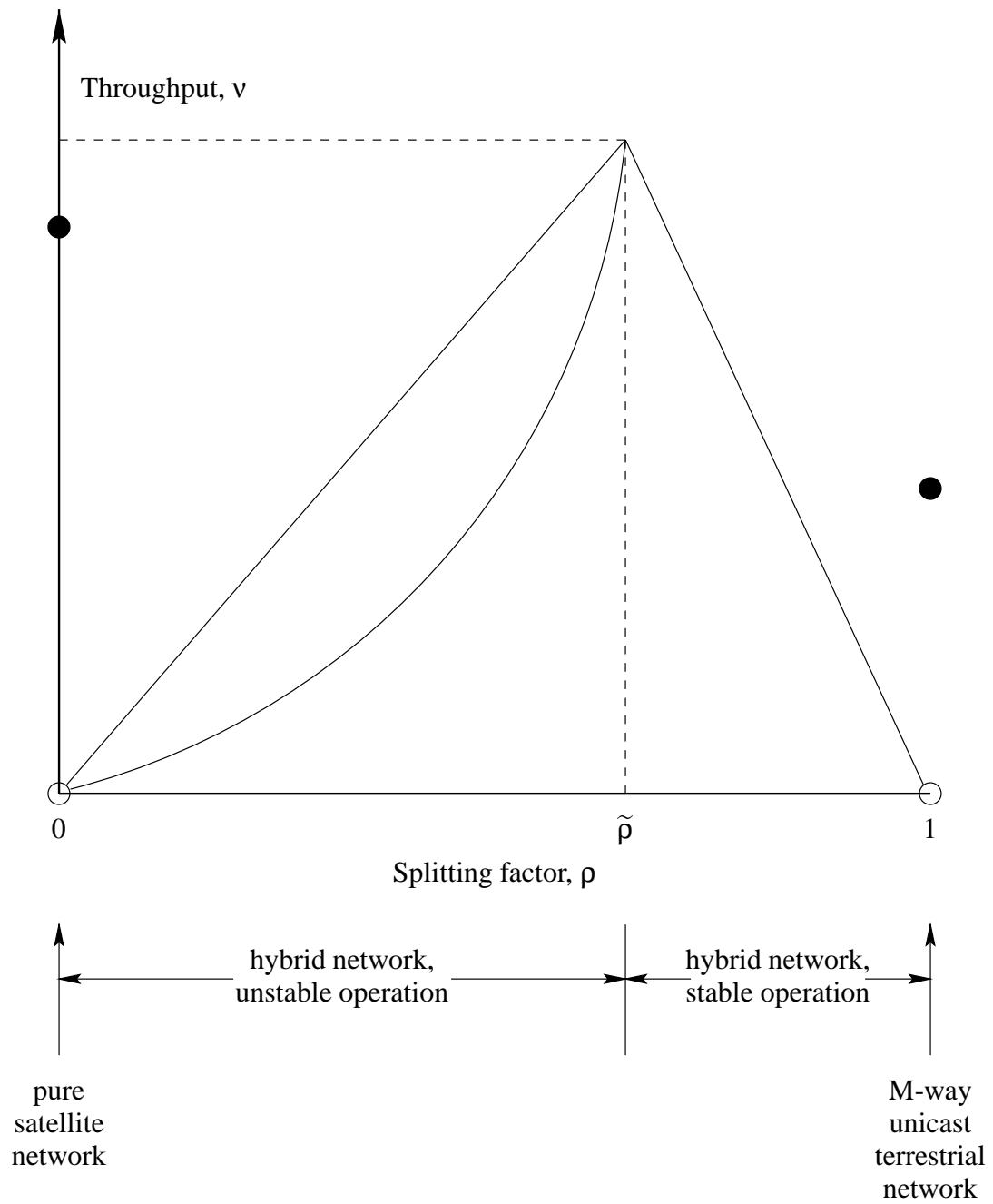


Figure 4.28: Sketch showing variation of throughput, v , with splitting factor, ρ .

is q_t instead of q_s . The throughput in this setting is then $\ell(1-p_t)r_{tot}/(\ell+h)M$. Thus the throughput is inversely related to M . Note that since there is no multicast link, transmissions to the M receivers need not be conducted simultaneously.

In the hybrid network, for which $0 < \rho < 1$, if too little terrestrial bandwidth is provided, then unstable operation will result. If sufficient terrestrial bandwidth is provided, then the operation will be stable. This suggests there is some value of ρ , which will be denoted here as $\tilde{\rho}$, which demarcates unstable and stable operation. More precisely, let $\tilde{\rho}$, $0 < \tilde{\rho} < 1$, be the maximum value of ρ which provides unstable operation. This particular definition for $\tilde{\rho}$ is asserted since ρ must actually exceed $\tilde{\rho}$ to assure stable operation.

So, if $\rho \leq \tilde{\rho}$, the hybrid network's operation will be unstable. As $\rho < \tilde{\rho}$ is reduced toward zero, bandwidth is taken from the terrestrial links and provided to the satellite link. This exacerbates the unstable character of the operation: a greater retransmissions load is provided to the terrestrial subsystem, while the service capability for this load is reduced. Hence the satellite transmitter idles for longer periods, and the throughput declines towards zero. As ρ is reduced, the change in retransmissions load (an increase) and the change of terrestrial capacity to service this load (a decrease) are linear changes. This suggests that the throughput decreases linearly as ρ is reduced from $\tilde{\rho}$ to zero. However, simulation results such as those presented in Figure 4.16 suggest that throughput may decline according to a convex curve as the degree of instability increases. Since it is not clear if the throughput varies linearly or according to a convex curve for $0 < \rho < \tilde{\rho}$, both possibilities are sketched in Figure 4.28.

If $\tilde{\rho} < \rho < 1$, then the hybrid network's operation will be stable. Adding terrestrial bandwidth will not improve throughput, since sufficient bandwidth has already been furnished to accommodate the retransmissions load. Increasing ρ reduces the satellite

link bandwidth, which reduces the retransmission load for the terrestrial subsystem. This only improves the stable character of the network's operation. Increasing ρ also reduces the rate at which frames are successfully delivered via satellite to a given receiver. So, increasing ρ beyond $\tilde{\rho}$ toward unity reduces the throughput toward zero. Since the throughput in a stable hybrid network is linear in the amount of satellite bandwidth, this reduction in throughput is also linear in ρ .

Based on these observations, the maximum throughput for the hybrid network is achieved with ρ slightly greater than $\tilde{\rho}$, and the throughput is then slightly less than $\ell\tilde{\rho}r_{tot}/(\ell+h)$.

To find $\tilde{\rho}$, writing the stability condition (3.4) in terms of ρ yields:

$$(1-\rho)r_{tot} \left[1 - (1-q_s)^{\ell+h} \right] < \frac{\rho}{M} r_{tot} (1-q_t)^{\ell+h} \quad (4.1)$$

Solving for ρ yields $\tilde{\rho}$:

$$\rho > \tilde{\rho} = \frac{1 - (1-q_s)^{\ell+h}}{\frac{1}{M}(1-q_t)^{\ell+h} + 1 - (1-q_s)^{\ell+h}} = \frac{p_s}{\frac{1}{M}(1-p_t) + p_s}, \quad (4.2)$$

where the last expression merely changes notation from BERs to FERs. This expression also indicates that $\tilde{\rho}$ increases with M , which is sensible: more receivers less bandwidth should be provided to the satellite link, and more should be provided to terrestrial links for retransmissions.

4.2.2.2 Variable Packet Length

If the packet length ℓ is not fixed, then ℓ can be adjusted to maximize throughput for a combination of q_s , q_t , $r_s = (1-\rho)r_{tot}$, and $r_t = \rho r_{tot}$, for each of the three cases of $\rho = 0$, $0 < \rho < 1$ and $\rho = 1$.

The optimal packet length for the $\rho = 0$ (pure-satellite network) case, ℓ_{sat}^* is calculated as described in Figure 3.4. With ℓ_{sat}^* so computed, p_s and then $\beta_{sat,M}$ can be found as well, so the throughput is $v_{sat,M} = \ell_{sat}^* r_{tot} / (\ell_{sat}^* + h) \beta_{sat,M}$.

If $\rho = 1$, then no satellite link is employed, and there is no multicasting. The optimal packet length in this setting, ℓ_{terr}^* , can be calculated as prescribed in (3.2) with q_t substituted for q_s . After computing p_t by (3.3), the throughput in this setting of M receivers is then $v_{terr,M} = \ell_{terr}^* (1 - p_t) r_{tot} / M(\ell_{terr}^* + h)$.

For the hybrid network, discussion in section 3.3.1.2 explains that idling of the satellite transmitter is guaranteed if the hybrid network's operation is unstable, and such idling significantly depresses throughput. Simulation results presented above verify this assertion and also demonstrate that such throughput degradation may be overcome by adjusting the packet length to achieve stable operation. It is clear, then, that if a finite window is employed for multicasting in a hybrid network, maximal throughput will be obtained for some combination of ℓ and ρ which yields stable operation.

It will be helpful to consider separately the cases of $0 < \rho \leq \tilde{\rho}$ and $\tilde{\rho} < \rho < 1$. For $\tilde{\rho} < \rho < 1$, namely stable hybrid operation, Figure 3.6 and (4.2) suggest the procedure for finding the best combination of packet length and splitting factor. Let ℓ_{hyb}^* denote the value of ℓ which sets to zero the derivative $\frac{\partial}{\partial \ell} \left\{ \frac{\ell+h}{\ell} \beta_{hyb,M} \right\}$. This value of ℓ will yield stable operation. Note that the derivative is a function of h , q_s , and q_t , but not of r_s or r_t . That is, ℓ_{hyb}^* is determined without regard to a particular value of ρ . Hence, for $\tilde{\rho} < \rho < 1$, the optimal packet length is ℓ_{hyb}^* . From Figure 4.28 it is clear that throughput is maximized for $\rho = \tilde{\rho}$ corresponding to ℓ_{hyb}^* .

The throughput in this case ($\tilde{\rho} < \rho < 1$) would then be

$$v_{hyb,M} \leq \frac{\ell_{hyb}^* r_{tot}}{\ell_{hyb}^* + h} \left[\frac{\frac{1}{M} (1 - p_t)}{\frac{1}{M} (1 - p_t) + p_s} \right],$$

where p_s and p_t are the FERs calculated using $\ell = \ell_{hyb}^*$. The expression is shown as an inequality because ℓ would have to be reduced to slightly less than ℓ_{hyb}^* to assure stability. A legitimate alternative would be to instead adjust ρ to be slightly greater than $\tilde{\rho}$.

So far it has been shown that the optimal splitting factor for $0 < \rho < 1$ is no greater than $\tilde{\rho}$ corresponding to ℓ_{hyb}^* . For $0 < \rho \leq \tilde{\rho}$, the network can still be used, but ℓ should be set to less than ℓ_{hyb}^* to assure stability by (4.1). Since there is no closed-form expression for each such ℓ , ρ must be varied on $(0, \tilde{\rho}]$, and for each value of ρ , the corresponding ℓ must be calculated, and then the throughput must be checked. It is quite possible the optimal ρ in $(0, 1)$ is less than $\tilde{\rho}$.

So, for $0 < \rho < 1$, the optimal combination of ρ and ℓ cannot be determined in a straightforward fashion as for $\rho = 0$ and $\rho = 1$. Instead, a search procedure on $(0, \tilde{\rho}]$ is required. The following summarizes the procedure:

1. Find ℓ_{hyb}^* , the value of ℓ which sets to zero the derivative $\frac{\partial}{\partial \ell} \left\{ \frac{\ell+h}{\ell} \beta(\ell) \right\}$.
2. Find p_s and p_t corresponding to ℓ_{hyb}^* , and then find $\tilde{\rho} = \frac{p_s}{\frac{1}{M}(1-p_t) + p_s}$.
3. Compute

$$\frac{\ell_{hyb}^* r_{tot}}{\ell_{hyb}^* + h} \left[\frac{\frac{1}{M}(1-p_t)}{\frac{1}{M}(1-p_t) + p_s} \right]$$

which is the maximum throughput achievable for stable hybrid operation, for which $\rho \in (\tilde{\rho}, 1)$.

4. Vary ρ over $(0, \tilde{\rho}]$; for each value of ρ , find the maximal value of ℓ which satisfies (4.1). Then use this value to compute the throughput $\ell(1-\rho)r_{tot}/(\ell+h)$.

5. The value of ρ , and the corresponding value of ℓ , which yield maximal throughput in the previous two steps is the combination which yields maximal throughput for $0 < \rho < 1$.

In particular, the maximal throughput may exceed $\ell_{hyb}^* (1 - \tilde{\rho}) r_{tot} / (\ell_{hyb}^* + h)$.

4.2.3 Summary

For the three cases of $\rho = 0$, $0 < \rho < 1$, and $\rho = 1$, the corresponding predicted optimal throughputs are:

$$\begin{aligned} \rho = 0 &\implies v_{sat,M} = \frac{\ell_{sat}^*}{(\ell_{sat}^* + h)} \frac{r_{tot}}{\beta_{sat,M}} \\ 0 < \rho < 1 &\implies v_{hyb,M} \approx \frac{\ell_{hyb}^* r_{tot}}{\ell_{hyb}^* + h} \left[\frac{\frac{1}{M} (1 - p_t)}{\frac{1}{M} (1 - p_t) + p_s} \right]; \\ &\quad \text{(use described search procedure)} \\ \rho = 1 &\implies v_{terr,M} = \frac{\ell_{terr}^*}{(\ell_{terr}^* + h)} \frac{(1 - p_t) r_{tot}}{M} \end{aligned}$$

Based on the given values of M , q_s , and q_t and r_{tot} , the optimal packet length and then the throughput can be computed for each of the three cases to determine which case provides the best throughput.

4.2.4 Numerical Example

The starting point chosen for a numerical example is the predicted throughput result shown for the hybrid network in Figure 4.26 with $q_s = 10^{-4}$, $q_t = 10^{-5}$, with $M = 5$ receivers assumed. In that setting, a total of $r_{tot} = 1.536 \text{ Mb/s} + 5 \times 33.6 \text{ kb/s} = 1.704 \text{ Mb/s}$ was used to achieve a predicted throughput of about 1.25 Mb/s. The corresponding splitting factor for that setting is $\rho = (5 \times 33600) / 1704000 = 0.098$.

The present task is to examine the variation of throughput as a function of splitting factor for the same link BERs, number of receivers, and total bandwidth.

For the particular values of q_s and q_t just given, $\tilde{\rho} = 0.2516$. On a graph showing throughput as function of ρ (such as Figure 4.28), which is the intended presentation format for the simulation results, this value of $\tilde{\rho}$ allows the range $0 < \rho < \tilde{\rho}$ to be seen easily. (The values for the BERs q_s and q_t for this example were selected to provide a value of $\tilde{\rho}$ which would be easily seen on such a graph.)

Using the specified values, the optimal packet length for $\rho = 0$ is 304 bits. The corresponding throughput is 1.29 Mb/s.

If $\rho = 1$, then the optimal packet length for operation with a common window is 908 bits, and 1980 bits for operating with independent windows. Rounding these packet lengths to nearest multiple of eight yields 912 and 1984 bits, respectively. The corresponding throughputs are approximately 310 and 328 kb/s, respectively.

For the hybrid network, the optimal packet length obtained by finding ℓ to set to zero the derivative $\frac{\partial}{\partial \ell} \left\{ \frac{\ell+h}{\ell} \beta_{hyb,M} \right\}$ is 651 bits. Rounding downward to a multiple of eight to assure stability and to suit simulation constraints yields 648 bits. The corresponding optimal value of ρ is $\tilde{\rho} = 0.2516$, as mentioned above, and the predicted throughput for $\rho = \tilde{\rho}$ is then 1.19 Mb/s.

Simulations were conducted to verify the throughput values as well as the supporting analysis presented above. The splitting parameter ρ was incrementally varied from zero to unity. For $0 < \rho \leq \tilde{\rho}$, simulations were conducted in two ways: with the 648-bit near-optimal packet length for $\tilde{\rho} < \rho < 1$, to examine the throughput in an unstable hybrid network; and with packet lengths computed for each value of ρ to assure stable operation.

For the $\rho = 1$ case, simulations were conducted both with and without a common

window for the five receivers.

For the simulations, some acknowledgements link bandwidth r_a had to be assumed. It was decided to set the acknowledgement link bandwidth r_a to equal the bandwidth of the link upon which frames were to be initially transmitted. That is, for $0 \leq \rho < 1$, r_a was set to $(1 - \rho)r_{tot}$, and for $\rho = 1$, r_a was set to r_{tot}/M . With r_a set so, and with the formulation of acknowledgements described earlier, it was expected that the acknowledgement link bandwidth would not constrain the throughput. This would allow considering the bandwidth splitting problem with negligible confounding effect of finite bandwidth for acknowledgements.

Earlier simulations indicated $\alpha_\theta = 0.25$ and $\epsilon_\omega = 10^{-5}$ yield throughputs which approach predicted values, and so these parameter values were used for the present group of simulations.

The simulation results are presented in Figure 4.29. (Predicted results are nearly coincident with the simulation results and are omitted for clarity.) These results match well the general prediction sketched in Figure 4.28. The results also indicate that a hybrid network's best performance is obtained at some particular $\rho < \tilde{\rho}$, with the packet length calculated to assure stability at that value of ρ . The figure indicates the best value for $\rho \in (0, 1)$ is about 0.15 for this scenario, with a corresponding packet length of 312 bits. As ρ is reduced from this point, the packet length required to assure stability quickly diminishes and the fraction of the frame occupied by the fixed amount of overhead packet quickly rises.

Overall, the pure-satellite network was found to provide the best throughput for the given combination of link BERs, total amount of available bandwidth, and number of receivers. Yet, if the number of receivers were to be increased, the throughputs obtained for $\rho = 0$ and $\rho = 1$ would decrease, while the throughput for the hybrid

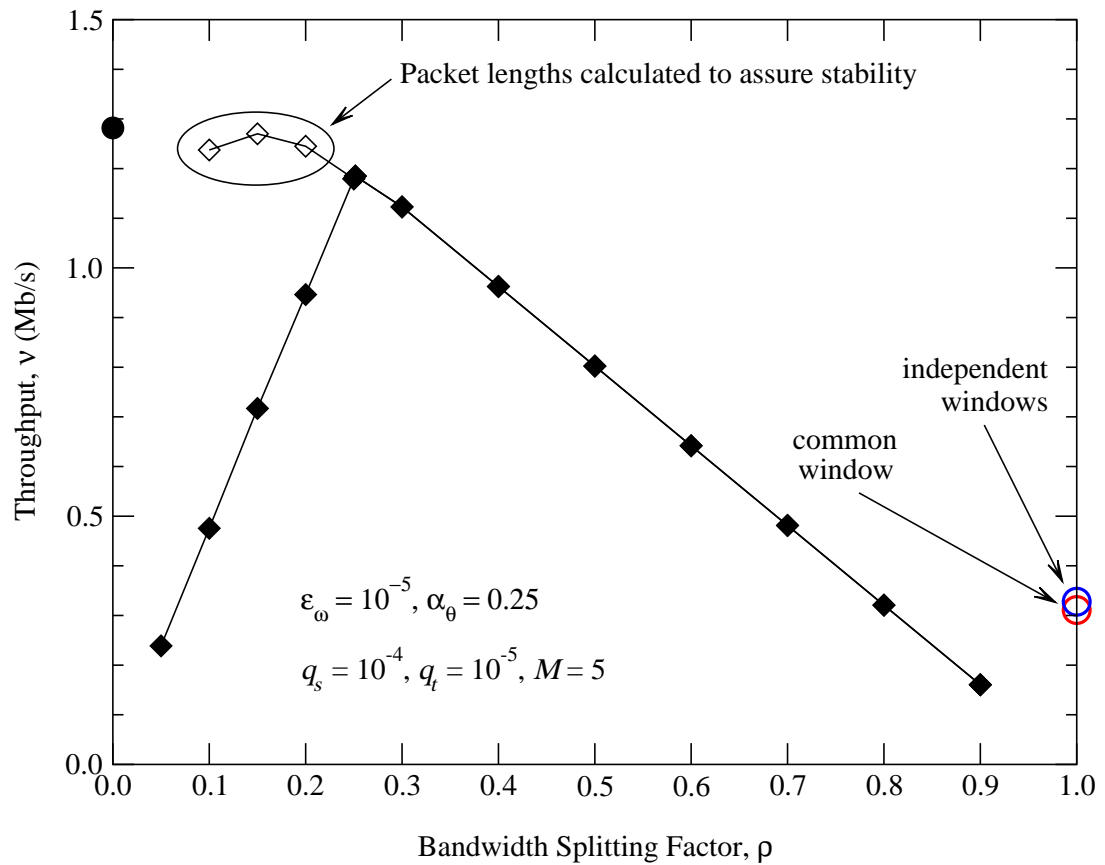


Figure 4.29: Results for example examining bandwidth splitting. For the hybrid network ($0 < \rho < 1$), a fixed packet length of 648 bits was used, except where noted.

network—in particular, for $\rho \approx 0.15$ —would not diminish. So, in a network having sufficiently many receivers, the hybrid network with $\rho \approx 0.15$ would provide the best throughput.

4.3 Multicasting to Non-Identical Receivers: The “Poor Listener” Problem

Until now, this dissertation has assumed the noise processes affecting reception are independent but identical at all receivers. Under such circumstances, and if the satellite link is not extraordinarily poor, the hybrid network was shown to be capable of alleviating the deterioration of throughput with the number of receivers in a satellite-based multicast network.

It is natural to wonder about performance when one receiver suffers exceptionally poor reception conditions. With an ideal multicast SR ARQ protocol, namely one with an unlimited window size, such a “poorer listener” would not affect throughput to other receivers in a hybrid network. However, with a common, finite window governing transmission to all receivers, the window’s advance would expectedly be limited by the rate at which frames can be successfully delivered to the exceptional receiver. Hence the throughput for all receivers would be limited by the poorer listener [4].

To explore this “poor listener problem,” assume a pure-satellite network has been engineered for M receivers to suit a given combination of r_s and q_s . In particular, the packet length ℓ has been determined, and there is a corresponding satellite link FER p_s . Suppose there are some M' exceptional receivers in the network which unfortunately experience a poorer satellite link, with BER $q_s' > q_s$. Consequently, the satellite link FER experienced by the exceptional receivers is some $p_s' > p_s$. To allow greatest

generality, let $0 \leq M' \leq M$. This range for M' implies a majority of receivers may be “exceptional,” but this terminology has clear intent and so will be retained. The other $M - M'$ receivers will be called the “ordinary” receivers.

The present interest is comparing the throughput of such a pure-satellite network with the throughput of an analogous hybrid network. More precisely, the hybrid network has been designed for M receivers to suit a given combination of r_s , r_t , q_s , and q_t , in which the M' exceptional receivers suffer with $q_s' > q_s$ (and so the link FER is worse as well, $p_s' > p_s$). To be clear, the packet length is calculated differently in the pure-satellite and hybrid networks, and so the satellite link FERs p_s and p_s' both differ in the two networks.

In calculating the throughput for multicasting in a pure-satellite network (section 2.2.1), the quantity $\gamma(j)$ was defined as the probability with which a frame is successfully delivered to all M receivers with j or fewer transmissions. In the poor listeners situation, $\gamma(j)$ retains its earlier definition, but must be calculated to reflect the differ FERs:

$$\gamma(j) = (1 - p_s^j)^{M-M'} \left(1 - (p_s')^j\right)^{M'}$$

The throughput is then calculated in the same fashion as before, namely

$$\beta = \sum_{j=1}^{\infty} j[\gamma(j) - \gamma(j-1)]$$

and

$$v_{sat} = \left(\frac{\ell}{\ell + h}\right) \frac{1}{\beta}.$$

As before, this throughput expression assumes an unlimited window size, and so may overestimate the throughput actually achievable.

In the hybrid network, the operation serving the ordinary receivers qualifies as stable, by design. If the operation serving the exceptional receivers also qualifies as stable, then the entire network's operation will be stable. The throughput to each of all M receivers would then be $\ell r_s / (\ell + h)$. If the operation serving the exceptional receiver(s) is unstable, and if the window size is unlimited, then the throughput for the network will be the average of the $M - M'$ stable operation throughputs and the M' unstable operation throughputs. So:

$$v_{\text{hyb}} = \begin{cases} \frac{\ell}{\ell+h} r_s, & \text{if operation is stable for all receivers} \\ \frac{\ell}{(\ell+h)M} \{ (M - M') r_s + M' [r_s (1 - p_s') + r_t (1 - p_t)] \}, & \text{if operation is unstable for the } M' \text{ receivers} \end{cases}$$

However, results and discussion presented earlier indicate that if operation serving a receiver is unstable, then a finite window will be exhausted at times. This will cause the satellite transmitter to idle, and so the throughput for that receiver will be far less than calculated. Hence the throughput for the case of the hybrid network serving M' exceptional receivers in unstable fashion cannot be calculated accurately either, and so the second case of the foregoing throughput expression is practically useful only as an upper bound, and this bound will likely be a loose one. If indeed operation is unstable for the M' receivers, then again throughput will be limited to all receivers in the network.

If $q_s \ll q_s'$, then the packet length calculated for the ordinary receivers may be large enough to yield unstable operation for the exceptional receivers. If q_s is reduced, then a greater value for ℓ will be calculated, and so p_s' will be increased if q_s' is held fixed. Hence, the better the satellite links to the ordinary receivers, the worse the performance will be for the exceptional receivers. In general, then, it is fair to assert

the following:

If a hybrid network is engineered for the ordinary receivers, and if the operation serving the exceptional receivers is unstable, then the greater the disparity between q_s and q_s' , the poorer the throughput to all receivers. So, the poor listener problem of the pure-satellite network is eliminated with the hybrid network only if the retransmission links have sufficient bandwidth to allow stable operation to all receivers.

A numerical example will now be introduced to aid further discussion. Assume $q_s' = 10^{-4}$, $q_t = 10^{-5}$, and, as in section 4.1, $r_s = 1.536$ Mb/s, $r_t = 33.6$ kb/s, and $h = 40$ bits. If $q_s = 10^{-7}$, then the optimal packet length is 5976 bits. The corresponding link FERs for the ordinary receivers are approximately $p_s = 6.0 \times 10^{-3}$ and $p_t = 5.8 \times 10^{-2}$. Checking the stability condition (3.4) yields the true relation $9213 < 31638$, verifying stable operation for the ordinary receivers. However, for the exceptional receivers suffering with $q_s' = 10^{-4}$, the corresponding p_s' is 0.45, and (3.4) yields the untrue relation $69440 < 31638$, indicating unstable operation for the exceptional receivers.

Suppose more generally that, for a given r_s , r_t , and q_t , the suboptimal procedure of setting $\ell = \lceil \tilde{\ell} \rceil$ to assure stability (as prescribed in Figure 3.6) is required for all q_s exceeding some value \hat{q}_s . If $\hat{q}_s \leq q_s'$, then the maximum packet length which will yield stable operation for the exceptional receivers is some $\ell' = \lceil \tilde{\ell}' \rceil$, where $\tilde{\ell}'$ denotes the value of ℓ which sets equal the LHS and RHS of (3.4) using q_s' instead of q_s . If $q_s < \hat{q}_s < q_s'$, the value of ℓ is calculated for the ordinary receivers by setting to zero the derivative $\frac{\partial}{\partial \ell} \left\{ \frac{\ell+h}{\ell} \beta(\ell) \right\}$. This ℓ exceeds the value of ℓ' , which is calculated from the stability condition since $\hat{q}_s \leq q_s'$. Since ℓ' is essentially the maximum acceptable packet length for stable operation for the exceptional receivers, but the network uses

packets of length $\ell > \ell'$, unstable operation may be expected for the exceptional receivers. In the foregoing numerical example, the intermediate satellite link BER value \hat{q}_s is approximately 10^{-5} . It is not surprising, then, that for $q_s' = 10^{-4} > \hat{q}_s = 10^{-5}$ and $q_s = 10^{-6} \leq \hat{q}_s$, the exceptional receivers were found to operate in unstable fashion.

If $q_s' < \hat{q}_s$, then stable operation may be possible, but is still not assured. In such a situation, the optimal packet length for the exceptional receivers, ℓ' , will be less than the transition packet length $\tilde{\ell}'$ but the difference may be small or large. If the packet length for the network, ℓ , satisfies $\ell' < \ell < \tilde{\ell}'$, then operation will be stable for the exceptional receivers (as well as for the ordinary ones). However, if $\tilde{\ell}' < \ell$, then operation will be unstable for the exceptional receivers, and throughput for all receivers will suffer.

To continue the foregoing numerical example, suppose again $q_s = 10^{-7}$, but $q_s' = 10^{-6}$. In this case, $\ell = 16984$, $\tilde{\ell}' = 18331$, and $\ell' = 5976$. The inequality sequence $\ell' < \ell < \tilde{\ell}'$ is satisfied in this case, and all receivers enjoy stable operation.

The foregoing numerical example was investigated with simulations of a network having $M = 5$ receivers. The results are shown in Figure 4.30. For $q_s' = 10^{-6}$, only one order of magnitude poorer than the satellite channel BER of $q_s = 10^{-7}$ for which the network was designed, it was shown above that this case yields stable operation in the hybrid network for the exceptional receivers, and so the simulated and predicted results are very similar for both the pure-satellite and hybrid networks. In the $q_s' = 10^{-4}$ case, the throughput is substantially poorer than predicted, and a drastic throughput reduction results if even only one receiver is exceptional. In this case, q_s and q_s' differ by three orders of magnitude, so the effect of the finite common window having been not regarded in the throughput prediction is clearly seen. Also, the exceptional receivers in the hybrid network suffer unstable operation in this case, and,

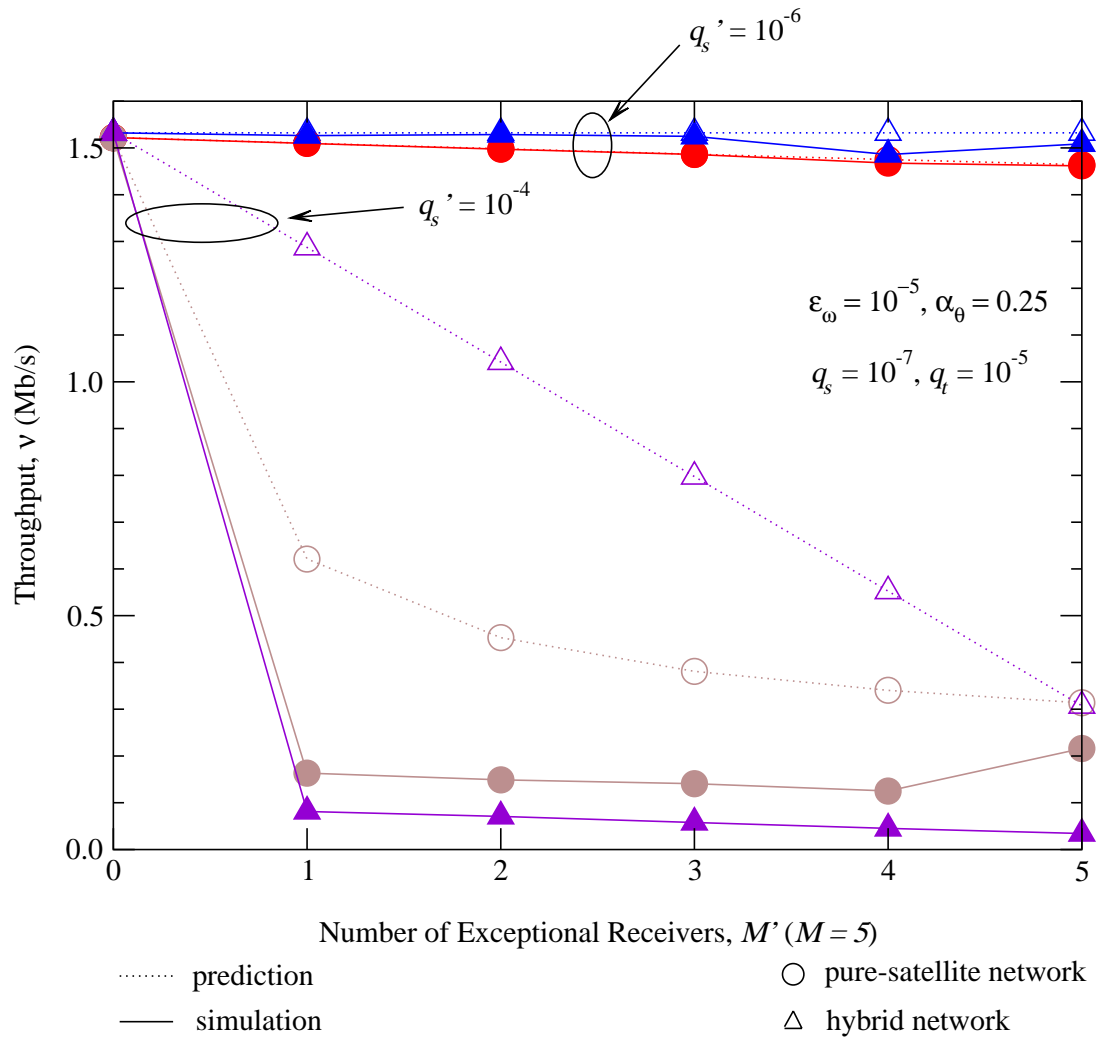


Figure 4.30: Throughput for “poor-listener” scenarios.

as asserted above, the throughput to all the receivers consequently diminishes greatly.

Chapter 5

Conclusion

5.1 Dissertation Review

This dissertation began by mentioning the superiority of ARQ, as an adaptive error control technique, for multicast error control. Immediately a problem inherent in this scheme was recognized: retransmissions typically benefit few receivers, and so the throughput diminishes with the number of receivers. The hybrid network was introduced to possibly overcome this problem, and analysis verifying this possibility was presented. Simulation was required for investigating further ARQ multicasting in pure-satellite and hybrid networks, so protocol implementation parameters were defined. The performance effects were considered for each of these parameters, and a methodology for selecting their values was developed. This methodology was used to develop simulations which indicated the throughput variation exerted by the parameters. Simulations also indicated how crucially the hybrid network must be operated in a “stable” fashion to realize good performance, and that in some network conditions such a stable hybrid network can indeed alleviate the loss of throughput seen in a multicast pure-satellite network. The optimal way to split a given fixed amount

of bandwidth between satellite and terrestrial links was considered, and the hybrid network was also shown to mitigate the poor listener problem in some circumstances as well.

5.2 Issues for Future Study

Perhaps foremost among matters for future examination is how to increase the efficiency of the protocol's operation in the hybrid network. At times when the satellite link experienced by one receiver is good that receiver's terrestrial channel may be underutilized. This suggests that if the transmitter can deduce the approximate quality of the satellite link to each receiver, perhaps initial transmissions of some frames can be conducted on the terrestrial links to increase the throughput and efficiency.

A related matter is how the acknowledgements might be composed differently to communicate more receiver buffer state information to the transmitter. While this is important for ARQ in general, it is particularly significant for a hybrid network with a severe constraint on acknowledgement bandwidth.

One matter requiring further investigation is how the combination of finite acknowledgement link bandwidth and limited acknowledgement length affects throughput. Although protocol analysis was disappointingly found to be intractable in several circumstances, experience with simulations provided some additional intuitive insights. Although an acknowledgement with limited length may not allow specifying all frames which must be retransmitted, timer expirations effect retransmissions for such unspecified frames. Hence excluding some numbers from an acknowledgement naklist of limited length may not be a serious concern. However, a concern persists that timers may initiate retransmissions for frames which were actually re-

ceived correctly but which could not be implicitly specified in the naklist. Although calculating the exact number of such frames is essentially impossible as it is for so many other intriguing conundrums of protocol operation, an approximate analysis may be possible.

Another surviving question is how the throughput delivered by an unstable hybrid network might be calculated. Although the poor throughput observed in simulations of such networks is a consequence of a finite window, again an approximate approach which circumvents the intractability of an exact analysis may be possible.

This dissertation examined fairly small numbers of receivers, to suit simulation constraints. In practice, thousands of receivers, or more, might be served by a multicast network. Hence, comparing the throughputs of pure-satellite and hybrid networks with such great numbers of receivers would be indicated before deploying satellite-based multicast networks of these sorts.

In this work, only a star topology of terrestrial links was considered; a natural extension is to consider other topologies. In particular, suppose the terrestrial network is a tree of terrestrial links, with the transmitter at the root node and a receiver at each non-root node. Such a tree could not only support multicasting in a hybrid network as described above, but would also allow a frame retransmission request sent by one receiver node to be serviced by the nearest ancestor node which is the root of a subtree having at least one node which successfully received the specified frame. The transmitter's load in servicing retransmission requests would then be reduced.

Such operation in a tree-shaped terrestrial network is similar to the operation of the Reliable Multicast Transport Protocol [29, 30]. Hence an integration of the hybrid network into multicasting development efforts by the Internet Engineering Task Force (IETF) is a possibility as well.

Tree-shaped terrestrial networks may also be wireless networks, as in the case of mobile receiving nodes. For example, mobile receivers, with omnidirectional antennas, can broadcast retransmission requests to other receivers possibly nearby and receive frames over the terrestrial wireless channel. A terrestrial tree for retransmissions, albeit a continuously changing tree, is perhaps applicable for mobile receivers as well.

A challenging concern for satellite multicasting to stations which are widely separated is that the stations may have significantly different satellite link FERs. In particular, weather phenomena may cause receivers in one region to similarly suffer a poor satellite link FER, while receivers in another region enjoy clear sky and correspondingly good communication. While this dissertation considered the “poor listener” problem to a limited extent, solutions for large networks with a large range of p_s values remain to be devised. One such solution might employ a tree-shaped terrestrial network logical topology, as described above, with subtrees defined on a regional basis to group together receivers with similar satellite channel FERs.

Additionally, frame error models more realistic than the BSC model considered in this dissertation remain to be investigated.

Hybrid ARQ schemes for multicasting, which employ FEC techniques for improving throughput, were mentioned in the Introduction. Such schemes suggest possibilities in the context of hybrid networks, such as using different FEC codes for the satellite and terrestrial channels. While this dissertation has explored hybrid networks, there are clearly a variety of additional matters open for further inquiry.

Appendix A

Useful Identities

For reference, some identities useful for this work are offered here. These identities apply for $0 < x < 1$ and $N \in \{0, 1, \dots\}$.

$$\sum_{i=0}^{\infty} x^i = \frac{1}{1-x} \quad (\text{A.1})$$

$$\sum_{i=0}^N x^i = \frac{1-x^{N+1}}{1-x} \quad (\text{A.2})$$

$$\sum_{i=0}^{\infty} ix^i = \frac{x}{(1-x)^2} \quad (\text{A.3})$$

$$\sum_{i=0}^{\infty} ix^{i-1} = \frac{1}{(1-x)^2} \quad (\text{A.4})$$

$$\sum_{i=0}^{\infty} i^2 x^i = \frac{x+x^2}{(1-x)^3} \quad (\text{A.5})$$

(A.1) yields:

$$\sum_{i=1}^{\infty} x^i = \frac{x}{1-x} \quad (\text{A.6})$$

(A.2) yields:

$$\begin{aligned}\sum_{i=1}^N x^i &= \frac{1-x^{N+1}}{1-x} - 1 \\ &= \frac{x-x^{N+1}}{1-x} = \frac{x(1-x^N)}{1-x}\end{aligned}\tag{A.7}$$

$$\begin{aligned}\sum_{i=2}^N x^{i-1} &= \sum_{i=1}^{N-1} x^i \\ &= \frac{x(1-x^{N-1})}{1-x} \\ \sum_{i=0}^N ix^{i-1} &= \frac{\partial}{\partial x} \sum_{i=0}^N x^i \\ &= \frac{\partial}{\partial x} \frac{1-x^{N+1}}{1-x} \\ &= \frac{1-x^{N+1} - (N+1)(1-x)x^N}{1-x}\end{aligned}\tag{A.8}$$

(A.4) yields:

$$\begin{aligned}\sum_{i=2}^{\infty} ix^{i-2} &= \frac{1}{x} \left[\sum_{i=1}^{\infty} ix^{i-1} - 1 \right] \\ &= \frac{1}{x} \left[\frac{1}{(1-x)^2} - 1 \right] = \frac{1}{x} \left[\frac{1-(1-x)^2}{(1-x)^2} \right] \\ &= \frac{1}{x} \left[\frac{1-1+2x-x^2}{(1-x)^2} \right] = \frac{2-x}{(1-x)^2}\end{aligned}\tag{A.9}$$

The binomial formula:

$$(x+y)^N = \sum_{i=0}^N \binom{N}{i} x^i y^{N-i}\tag{A.10}$$

The mean of a binomially-distributed random variable is computed by:

$$\sum_{i=0}^N i \binom{N}{i} x^i (1-x)^{N-i} = Nx\tag{A.11}$$

(A.11) trivially yields:

$$\sum_{i=1}^N i \binom{N}{i} x^i (1-x)^{N-i} = Nx \quad (\text{A.12})$$

Appendix B

Glossary of Notation

Symbol	Meaning
Superscripts and Subscripts	
$(\cdot)'$	Quantity pertains to “exceptional” receiver(s)
$(\cdot)^*$	Quantity has been calculated to be optimal, i.e. to provide maximal throughput
$(\cdot)_a$	Quantity pertains to acknowledgements link
$(\cdot)_s$	Quantity pertains to satellite link
$(\cdot)_t$	Quantity pertains to terrestrial link
$(\cdot)_{hyb}$	Quantity pertains to hybrid network
$(\cdot)_{hyb}^s$	Quantity pertains to satellite link in hybrid network
$(\cdot)_{hyb}^t$	Quantity pertains to terrestrial link in hybrid network
$(\cdot)_{sat}$	Quantity pertains to pure-satellite network
$(\cdot)_{terr}$	Quantity pertains to terrestrial network

Symbol	Meaning
--------	---------

Greek Characters

β	Average number of frames transmitted per frame delivered to all receivers (a measure of <i>inefficiency</i>)
$\gamma(\cdot)$	Cumulative distribution function for the number of transmissions required to successfully deliver a frame to all receivers
ε	Arbitrarily chosen desired upper bound for probability of an undesirable event
η	Throughput efficiency (ratio of throughput to sum of link bandwidths used to achieve that throughput)
θ	Maximum number of sequence numbers an acknowledgement may contain ($\theta \in \{2, 3, \dots\}$)
λ	Arrival rate
ν	Throughput (information bits/second)
ρ	Fraction of total available bandwidth devoted to terrestrial links
τ	One-way, station-to-station propagation delay (seconds)
ϕ	Number of frame transmission intervals used for computing timer period ($\phi \in \{2, 3, \dots\}$)
ω	Maximum possible number of transmission attempts without exhausting the ARQ window ($\omega \in \{1, 2, \dots\}$; $\omega \geq 2$ for SR ARQ)

Symbol	Meaning
Roman Characters	
h	Number of overhead (non-information) bits per information frame sent, via either satellite or terrestrial link ($h = h_{CRC} + h_{seq}$)
h_{CRC}	Number of bits for cyclic redundancy check (CRC)
h_{seq}	Number of bits for [each] sequence number in an acknowledgement
K_{ack}^{max}	Maximum number of bits per acknowledgement frame ($K_{ack}^{max} = \theta h_{seq} + h_{CRC}$)
L	Total number of bits per frame ($L = \ell + h$)
ℓ	Number of information bits per frame
M	Number of receivers
N	ARQ window size (frames)
p	Frame-error probability (also called frame-error rate, FER)
q	Bit-error probability (also called bit-error rate, BER)
r	Bit transmission rate (bits/second)
T	Timer period (seconds)

Bibliography

- [1] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice Hall, 1983.
- [2] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*. Prentice-Hall, 1995.
- [3] T. Pratt and C. W. Bostian, *Satellite Communications*. Wiley, 1986.
- [4] Y. Yamauchi, “On the packet radio multicast scheme for the personal communications era,” in *International Conference on Communication Systems (ICCS '94), Singapore*, pp. 576–580, IEEE, 1994.
- [5] S. B. Calo and M. C. Easton, “A broadcast protocol for file transfer to multiple sites,” *IEEE Transactions on Communications*, vol. 29, pp. 1701–1707, November 1981.
- [6] K. Mase, T. Takenaka, H. Yamamoto, and M. Shinohara, “Go-back- N ARQ schemes for point-to-multipoint satellite communications,” *IEEE Transactions on Communications*, vol. COM-31, pp. 583–589, April 1983.
- [7] I. S. Gopal and J. M. Jaffe, “Point-to-multipoint communication over broadcast links,” *IEEE Transactions on Communications*, vol. 32, pp. 1034–1044, Sept. 1984.

- [8] K. Sabnani and M. Schwartz, "Multidestination protocols for satellite broadcast channels," *IEEE Transactions on Communications*, vol. 33, pp. 232–240, Mar. 1985.
- [9] R. H. Deng, "Hybrid ARQ schemes for point-to-multipoint communication over nonstationary broadcast channels," *IEEE Transactions on Communications*, vol. 41, pp. 1379–1387, Sept. 1993.
- [10] A. Shiozaki, "Adaptive type-II hybrid broadcast ARQ system," *IEEE Transactions on Communications*, vol. 44, pp. 420–422, April 1996.
- [11] H. Liu, Q. Zhang, M. E. Zarki, and S. Kassam, "Wireless video transmission with adaptive error control," in *1996 International Symposium on Information Theory and its Applications (ISITA '96), Victoria, British Columbia*, pp. 371–374, 1996.
- [12] H. Zhao, T. Sato, and I. Kimura, "A hybrid-ARQ protocol with optimal adaptive error control for multidestination satellite communications," in *International Conference on Communication Systems (ICCS '94), Singapore*, pp. 420–424, 1994.
- [13] J. L. Wang and J. A. Silvester, "Optimal adaptive multireceiver ARQ protocols," *IEEE Transactions on Communications*, vol. 41, pp. 1816–1829, Dec. 1993.
- [14] S. R. Chandran and S. Lin, "Selective-repeat-ARQ schemes for broadcast links," *IEEE Transactions on Communications*, vol. 40, pp. 12–19, January 1992.
- [15] M. A. Jolfaei, S. C. Martin, and J. Mattfeldt, "A new efficient selective repeat protocol for point-to-multipoint communication," in *IEEE International Conference on Communications (ICC '93)*, vol. 2, pp. 1113–1117, 1993.

- [16] U. Quernheim, R. Vermöhlen, and M. Aghadavoodi Jolfaei, “A new ARQ-scheme for multicast satellite communication,” in *3rd European Conference on Satellite Communications*, pp. 11–15, IEE, Nov. 1993. Conference Publication No. 381.
- [17] M. A. Jolfaei and U. Quernheim, “A new selective repeat ARQ scheme for multicast communication,” in *IFIP TC6 Second International Conference (“Broadband Communications II”)*, in *IFIP Transactions C [Communications Systems]*, vol. C-24, 2-4 March 1994.
- [18] K. Sabnani, *Multidestination Protocols for Satellite Broadcast Channels*. PhD thesis, Columbia University, 1982.
- [19] M. Schwartz, *Telecommunication Networks: Protocols, Modeling, and Analysis*. Addison-Wesley, 1987.
- [20] UCB/LBNL/VINT Network Network Simulator NS, <http://mash.cs.berkeley.edu/ns>, March 1999. (NS is an open-source simulator, developed and maintained by the VINT [Virtual InterNetwork Testbed] Project, a collaboration among the University of Southern California/Information Sciences Institute, Xerox Palo Alto Research Center, Lawrence Berkeley National Laboratory, and the University of California at Berkeley, with contributions received from others throughout the world via the Internet.).
- [21] International Standards Organization, *International Standard ISO 4335, “Information processing systems – Data communication – High-level data link control procedures – Consolidation of elements of procedures”*.

- [22] International Standards Organization, *Proposed Draft International Standard ISO 7776/DAM 2, "Information processing systems – Data communication – High-level data link control procedures – Description of the X.25 LAPB-compatible DTE data link procedures – DAM 2: Multi-selective Reject Option"*.
- [23] D. Bertsekas and R. Gallager, *Data Networks*. Prentice-Hall, 2nd ed., 1992.
- [24] F. Halsall, *Data Communications, Computer Networks, and Open Systems*. Addison-Wesley, 3rd ed., 1992.
- [25] European Telecommunications Standard Institute, *GSM 03.34 [Draft Version 0.43.0], "European digital cellular telecommunications system (Phase 2+); High Speed Circuit Switched Data (HSCSD); Stage 2 Service Description"*, February 1996.
- [26] European Telecommunications Standard Institute, *GSM 04.22 [Version 4.0.0], "Digital cellular telecommunications system (Phase 2+); Radio Link Protocol (RLP) for data and telematic services on the Mobile Station–Base Station System (MS-BSS) interface and the Base Station System–Mobile-services Switching Centre (BSS-MSC) interface"*, December 1995.
- [27] G. Maral and M. Bousquet, *Satellite Communications Systems: Systems, Techniques and Technology*. John Wiley & Sons, 1998.
- [28] D. Friedman, "Error control for satellite and hybrid communication networks," Master's thesis, University of Maryland at College Park, 1995.
- [29] S. Paul, K. K. Sabnani, J. C. Lin, and S. Bhattacharyya, "Reliable multicast transport protocol (RMTP)," *IEEE Journal on Selected Areas in Communications*, vol. 15, pp. 407–421, April 1997.

- [30] J. C. Lin and S. Paul, “RMTP: A reliable multicast transport protocol,” in *Proceedings of IEEE INFOCOM '96*, pp. 1414–1424, 1996.