

DESIGN OF DISTRIBUTED COMPUTER
- COMMUNICATION NETWORKS

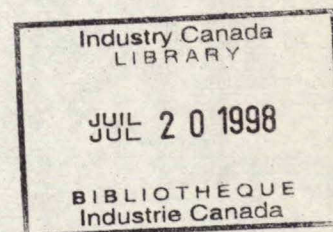
Robert W. Donaldson

P
91TH
C655
D6512
1977

P
91
C655
D6512
1977

2 DESIGN OF DISTRIBUTED
COMPUTER - COMMUNICATION NETWORKS

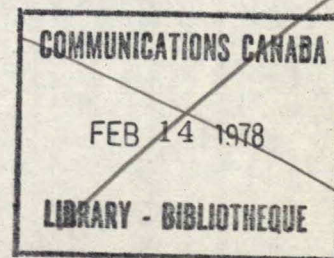
A Report to the
Department of Communications
Ottawa, Ontario, Canada



by

Dr. Robert W. Donaldson, Professor
Department of Electrical Engineering
University of British Columbia
Vancouver, British Columbia
Canada

July 1977



ABSTRACT

This report deals with the design of networks for support of various electronic information services, including word processing and office automation, electronic mail, banking and electronic funds transfer, automated reservation systems, teleconferencing, computer aided instruction including via home terminals, health care, library and other information systems, automated utility meter reading and computer-to-computer communications.

The network design problem is formulated as a constrained optimization of network cost, delay or reliability with respect to one or more of data link capacity, link flow or topology. Source-to-destination traffic and component costs are specified at the outset, although it is recognized that these are often inaccurate estimates subject to rapid change. Exact design solutions may be obtained, but only if the topology is fixed at the outset.

Message queuing at network nodes and various multiplexing schemes are considered in detail, and their affect on network performance is established.

Network operation and management, the area of our greatest ignorance, is considered under the following specific topic headings: data link control, data link error control, source-destination control, priority queuing, adaptive routing, flow control, buffer management and radio network protocols.

The need for additional study is indicated with respect to the following specific topics: improved estimates of cost and traffic trends; incorporation of our (or some other) proposed reliability measure as a formal design constraint; further study of modified contention multiplexing schemes; development of topological design methods which generate efficient hierarchical networks; improved knowledge of minimal required overheads as well as ways of achieving these for both data link controls and source-to-destination controls; improved understanding and realization of adaptive routing and flow control as well as nodal and destination buffer design and management; study of radio network routing problems; and serious study of distributed data base design, interprocess communication and internetworking.

TABLE OF CONTENTS

	Page
I. INTRODUCTION	
I-1 Computer Communication Systems and Services	1
I-2 Major System Factors	3
I-3 Outline and Summary of the Report	5
I-4 Recommendations for Further Study	14
I-5 References	17
II. FORMULATION OF THE NETWORK DESIGN PROBLEM	
II-1 Representative Electronic Information Service (EIS) Applications	22
II-2 Network Message Statistics	25
II-3 Network Costs and Performance Criteria	29
II-4 Network Design Problem Statement	35
II-5 Circuit, Message and Packet Switching	42
II-6 References	46
III. MESSAGE QUEUING AND CHANNEL SHARING	
III-1 Introduction and Overview	53
III-2 Modems	54
III-3 Message Queuing at Network Nodes	59
III-4 Non-Contention Multiplexing	67
III-5 Contention Multiplexing	74
III-6 Modified Contention Multiplexing	79
III-7 Contention Multiplexing and Queuing Interactions	86
III-8 Polling	89
III-9 References	92

IV. NETWORK DESIGN

IV-1	Network Design; Variables and Constraints	97
IV-2	Data Link Capacity Optimization	98
IV-3	Traffic Flow Assignment (Fixed Routing) .	106
IV-4	Flow and Capacity Assignment	110
IV-5	Topological Design of Centralized Networks using Graph Heuristics	111
IV-6	Topological Design of Centralized Networks using Multidropped Lines	116
IV-7	Topological Design of Centralized Networks via Nodal Clustering	117
IV-8	Topological Design of Distributed Networks: Non-Clustering Approaches	123
IV-9	Location of Backbone Nodes	126
IV-10	Topological Design of Distributed Networks using Nodal Clustering	128
IV-11	Effects of Traffic Level on Network Costs	153
IV-12	Unresolved Problems in Network Design . .	134
IV-13	References	136

V. OPERATION AND MANAGEMENT OF NETWORKS; NETWORK PROTOCOLS

V-1	Network Management Issues	139
V-2	Data Link Controls	140
V-3	Data Link Errors and Error Control	147
V-4	Priority Queuing	153
V-5	Adaptive Routing	159
V-6	Flow Control	166
V-7	Nodal Buffer Design and Management	175
V-8	Source-Destination Controls	179
V-9	Protocols for Radio Networks	183
V-10	References	187

LIST OF FIGURES

	Page
Fig. 2-1 Gamma distribution (a) Density function $f_x(U)$ (b) Moment ratio	28
Fig. 2-2 Typical electronic information network . . .	37
Fig. 3-1 Model of a physical communication channel .	56
Fig. 3-2 Queuing of messages at a network node . . .	60
Fig. 3-3 \bar{x} , $\overline{x^2}$ and ϕ for various message length distributions	64
Fig. 3-4 Waiting time W_e/\bar{x} and delay T_e/\bar{x} for exponential message lengths	66
Fig. 3-5 Illustrating FDMA and TDMA (a) M separate channels of capacity C/M (b) Single shared channel of capacity C	69
Fig. 3-6 Parameters a(M) and b(M)	78
Fig. 3-7 Maximum throughput for various multiplexing schemes; a = 0.01 [K2]	83
Fig. 4-1 Illustrating hierarchical routing	131
Fig. 5-1 Standardized word (frame) format for data link control	144
Fig. 5-2 Illustrating round-robin scheduling	158
Fig. 5-3 Simplified model for flow control analysis .	172

I. INTRODUCTION

I-1 Computer-Communication Systems and Services

It is now cost-effective to interconnect geographically separated computers and user terminals for purposes of sharing hardware, software, and data, and for purposes of terminal-to-terminal communications [K1, K2, R1, S1]. Thus, users are potentially able to communicate with each other and with various computers. Interactive data communication is now a reality whose growth will continue, and will lead in due course to a variety of electronic information services (EIS).

Services which either exist now or are potentially available include:

1. word processing and office automation [M1, O1, O2, F1];
2. electronic mail [H1, L1, M2, W1, R2];
3. banking and electronic funds transfer [J1, F2, S2, P2, M1];
4. reservation systems for airlines, hotels and other transportation and hospitality facilities [M1, M3];
5. teleconferencing [V1, L2, J2];
6. computer-aided instruction, including via home terminals [M1, B1; P3, H2, C1];
7. health care and hospital information systems [M1, K3, K4, R3, S3, S4];
8. remote accessing of information files detailing library books available, transportation schedules, weather forecasts, and coming events [M1, B1, C1, S5];
9. automated reading of utility meters [M1, B1].

These and other services are described in detail elsewhere as indicated by the above references.

In an earlier report [D1] it is argued that the cost of a specific electronic information service will be reduced if a variety

of related services are available to share various common costs. For example, the cost of supplying a single business establishment with a word processing system to assist in document preparation might be prohibitive. However, if an electronic mail service and an electronic funds transfer capability were also available, and if many business establishments were able to share these services, the per-unit costs of each to any one business might well be acceptable. Thus, the emergence of a number of different electronic information services over a relatively short time frame is likely. Although varying estimates as to the eventual size of the EIS industry are available [K5, P4], most agree that the industry will contribute significantly to the gross national product of our post industrial society.

Much of the work to date has involved fixed terminals serviced by terrestrial networks. However, there is growing interest in employing satellites as well as radio networks for use in remote areas or with mobile terminals [K6, R4].

The early work in interactive data communications involved communications between computers, and between expert users and computers [A1, K2, C2, S1, S6]. Because of the obvious extension of this earlier interest to a variety of applications, we consider the design of networks to support electronic information services, which include computer communications.

Actual networks, both operational and planned are described in a variety of references [S1, K2, V2, P1, C3, S6]. Schwartz [S6] includes a relatively brief but complete description of four representative networks selected from approximately a dozen different networks in North America, Europe and Australia.

I-2 Major System Factors

The broad array of EIS subject matter motivates subdivision of the relevant material into relatively disjoint, manageable pieces. Examination of the various services leads to isolation of the following four separate but related factors [C2, D1]:

1. computer-communication network analysis and design
2. man-machine interfacing
3. organization and management of distributed data bases
4. socio-economics.

The first item above is the subject of the present report. Involved here is the interconnection of computers and user terminals via a network whose topology, data link capacity and data link flow must be specified, along with network management procedures for controlling message flow in such a way that network performance as measured by message delay, reliability and throughput is cost-effective. A more rigorous definition of the network design problem is presented in Chapter 2.

The fact that many industrialized countries have or soon will implement networks for support of EIS services motivates the growing interest in internetworking, which involves communication between users or resources in different networks [R5, S7, C4]. The present report deals solely with intra-networking, where many problems still remain unsolved. Improved understanding of these problems, particularly network operation and management and network protocol design are essential to effective internetworking.

Man-machine interfacing involves two related matters:

1. conversion of text, speech, and images into electronic form either for "understanding" by machine (the pattern recognition or scene analysis problem) or for reconstruction

- at some later date (the source coding problem);
2. design of terminals to enable users to interact with other users, computers and data banks.

An earlier report [D1] provides a detailed summary of those efficiencies which can and cannot be achieved in coding speech, monochrome images, color images, and text. This same report provides an introduction to the design and evaluation of user terminals. The need for further work in terminal design is clearly indicated.

Data base design involves specification of data structures as well as procedures and arrangements for retrieving, storing, updating and processing data. Design goals include versatility, application independence and ease of maintenance. Versatility includes provision for obtaining answers to questions not specifically anticipated by the data-base designer. Virtually all design efforts have involved centralized rather than distributed data base configurations. A centralized data base stores [M4, M5, K7] all data at one site. Distributed data bases involve storage at several locations, in which case relationships between the data base and the network interconnecting these cannot be ignored [C5, B2, F3]. Now is an appropriate time to consider these interrelationships, in order to avoid errors that might prove costly or irreversible.

Socio-economics includes a multitude of difficult sociological considerations including security and privacy of information on the one hand, and public access to information and prevention of its unwarranted manipulation on the other. The relevant sociological aspects are discussed (but not fully resolved!)

in a variety of publications [M1, C1, P4, K5, V3, S8, A2].

Economic matters are considered in an earlier report [D1] under three subheadings:

1. cost of supplying electronic information services
2. demand for services
3. economic options for governments

Clear relationships exist between supply costs and the network design problem; specifically, the latter involves the arrangement of data links, concentrator-multiplexors, computers, and computer peripherals to provide a specified grade of service at minimum cost. There is also a clear relationship between the cost of transmitting a message and the number of bits required to encode the message; the source coding problem, therefore, relates to economics. Various literature references dealing with one or more aspects of EIS economics are available [B1, B3, C1, D2, H3, M6, N1, P4, P5, P6, R1, T1].

Finally, we list several references of a general nature which are recommended as further reading on various aspects of the network design problem [A1, K2, M7, V2, P1, W2, P7, C3, B1, B4, S1].

I-3 Outline and Summary of the Report

The following detailed summary replaces summaries which might otherwise conclude each chapter.

Chapter 2 deals with the formulation of the network design problem, as well as related matters which include representative EIS applications, traffic characteristics, component costs and cost trends, performance criteria and circuit, message and packet switching.

Even a brief examination of representative EIS applications indicates that traffic will consist of both file transfers involving messages of length from 10^4 to 10^7 bits and interactive conversational messages of a few hundred bits. Existing EIS traffic statistics are sparse, and are subject to the rapid and continuing changes characteristic of a growth industry. Existing traffic data tends to support existing mathematical models which employ Poisson message arrivals, geometric message lengths, and Gamma-distributed inter-character and inter-burst times.

Network performance criteria include average message delay, message traffic throughput and network reliability. Related to but separate from reliability are sensitivity of network performance to changes in design data and network adaptability to such changes.

Although the trend of decreasing real costs of communication links, memory and CPU's will likely continue, different rates of decrease are indicated for different items, and in the case of memory and CPU's, not easily forecast. Any decrease in software costs due to standardization might be offset by increases in progress complexity. These cost uncertainties provide obvious difficulties for those who must design on the basis of anticipated costs.

Because message delay is normally averaged over all network traffic, individual users may experience considerable variations about the average. Network management policies including message priority assignment strategies, flow control and adaptive routing are needed to reduce the effects of delay variations seen by individual users.

The fact that reliability is not normally included at the outset as a formal network design constraint is deemed a weakness in existing design procedures. A formal definition of reliability is proposed which represents the fraction of traffic unable to reach its destination because of link or node failures. Its linearity with respect to network flows facilitates its incorporation into existing analytical design procedures.

As proposed, the network design problem involves optimization of either cost, delay or reliability with respect to the network variables while the other two criteria are constrained. Sensitivity and adaptability is assessed by observing the effects on the performance criteria when design variables such as network traffic and network component cost are varied.

Because EIS applications involve both conversational and file traffic, packet switching seems preferable to either message or circuit switching. Unfortunately, a definitive comprehensive comparison between circuit message and packet switching is lacking, as is a viable analytical procedure for optimizing packet length. Existing evidence indicates that network performance is not strongly affected by packet length variations about the optimum.

Because messages are generated at irregular intervals and are of unpredictable length, queues form at network nodes. Chapter 3 presents results enabling calculation of delays experienced by single messages or packets which must queue for service. The effects of actual message length distribution on queuing delays are considered. For unimodal distributions, queuing delays are not strongly dependent on the actual length distribution.

Different message streams are often required to share a

physical channel, particularly where satellite or radio links are used. The remainder of Chapter 3 is devoted to considering various multiplexing schemes and their effects on message delay and throughput.

Frequency division or time division multiple access is seen to be poorly suited to EIS data traffic whose low-duty cycle and bursty character requires a high bandwidth channel at infrequent and random times. Spread-spectrum multiple access is suited to such traffic and warrants further evaluation.

Other multiplexing schemes include contention schemes such as pure and slotted ALOAH where a user accesses a shared channel with little or no regard for other channel users. "Collisions" between users necessitate retransmission, hopefully in a way which will avoid repeated collisions. Delay and throughput equations are presented for various multiplexing strategies, including those involving pure contention, partial contention, or no contention (roll-call polling). Each multiplexing scheme is favourable in a specific environment. Contention schemes are easily implemented and useful in light traffic situations. Modified contention schemes which result from unequal assignment of transmitter powers to various users, channel sensing, or dynamic channel reservations are better in heavy traffic situations. Polling is useful in heavy traffic when delay requirements or number of users are not too high. Delays which result from interactions of queuing and retransmissions caused by collisions are analyzed. The need for additional study and evaluation of modified contention schemes is indicated.

Chapter 3 also includes a brief discussion of modems, since

these are major determinants of the actual rate at which data can be transmitted at a specified error rate. It is noted that modem turn-around time arises from echo suppressor reversal and synchronization; each of these operations may cause 150 ms delays with the result that full-duplex rather than half-duplex transmission is a virtual necessity when delay requirements are stringent.

Chapter 4 provides a reasonably complete summary of existing network design procedures. Optimization of point-to-point link capacities can be readily achieved for both continuous and discrete cost-vs-capacity functions. Recent link optimization studies have included ALOAH multiplexed satellite links. Link optimization in modified contention multiplexing situation requires simplified approximate fits to delay-vs-throughput curves.

Optimization of (non-adaptive) link traffic flows is possible using flow deviation to obtain the (unique) optimum flows. Joint optimization of flows and capacities is more difficult, and involves random selection of initial solutions which will normally yield different local minima, from which the smallest is selected as best.

Various approaches are available for joint optimization of the topology, capacity and flow of centralized networks which may use either data concentrators or multidropped lines. Design algorithms which begin by partitioning network nodes into clusters followed by optimization of each cluster seem best.

Algorithms for designing distributed networks are available. Those which do not use clustering include branch exchange, concave branch elimination and cut saturation. The algorithms yield networks which differ considerably in topology but seem comparable in

cost and performance. One design approach involves partitioning of the design problem in local access network design and distributed network design. Backbone switch nodes interface the local networks to the distributed high-speed network. The approach simplifies the original design problem at the expense of an initial constraint on the network's hierarchical structure.

Recently, clustering has been employed in the design of singly-connected hierarchical distributed networks. Extension of the method to yield multiply-connected networks seems feasible. For networks with a large number (≈ 1000) of nodes, hierarchical networks are required to prevent prohibitive growth in the size of the nodal routing tables and routing update traffic. In such networks routing is via a set of gateway nodes in various hierarchical nodal clusters. Such a routing strategy increases message path length as well as message delay, but the increases seem to be more than offset by decreases in nodal buffer costs and delays, and reduced levels of routing update traffic.

Arguments are presented to show that as traffic level increases, cost/bit transmitted should decrease. The arguments are supported by results from actual design studies which, however, do not fully consider network operation and management costs, including traffic overhead.

Although several methods exist for network design, the best topological layout, link capacity assignment and traffic flow assignment, as well as the resulting optimum performance cannot normally be determined. Further study is needed to combine clustering approaches for distributed network design with hierarchical routing considerations. The fact that EIS network traffic

and node locations will change rapidly during the EIS industry's growth years motivates inclusion of sensitivity and adaptability measures in design procedures. The need for a formal reliability constraint was noted earlier.

Chapter 5 deals with the area of our greatest ignorance, namely network operation and management, and network protocols. Our subdivision of subject matter is somewhat arbitrary but convenient, and includes data link controls, data link errors and error control, priority queuing, adaptive routing, flow control, buffer design and management, source-destination controls and radio network protocols.

Data link controls (DLC's) enable node-to-node transmission of messages or packets by initiating transmission, terminating transmission, providing word (frame) synchronization and combating link errors. DLC's can be assessed in terms of efficiency, which is based on control overhead, and reliability, which involves the probability of successful node-to-node transmission. Although existing DLC's are reliable, they seem inefficient; 35 percent of a fully loaded ARPANET carries DLC information. The minimum required DLC information as well as ways of achieving this minimum has been considered, but only for highly idealized and rather unrealistic situations. Two important parameters for which optimization has been successfully considered are synchronization prefix length and ARQ retransmission period.

Statistical fluctuations in both short term and long term traffic as well as data link degradations motivate adaptive routing, which should improve network performance over that for fixed routing

since the latter does not fully exploit temporarily underutilized links. Both centralized and distributed adaptive routing algorithms are currently used in TYMNET and ARPANET, respectively. However, much remains to be learned regarding adaptive routing including the optimal division of centralized and distributed control, selection and updating of information on which to base routing decisions, routing algorithm development and assessment, and routing update overhead minimization.

Flow control includes those measures used to prevent an individual user or group of users from hoarding network resources. Virtually all flow control schemes permit entry of a packet to a network only if sufficient buffer storage is available and has been allocated somewhere along the source-destination route and rejects packets either initially or during transit if the buffer occupancy in either a portion or all of the network exceeds some threshold. A multitude of strategies consistent with these two constraints exist, and analysis of these various strategies seems very difficult. A general program recently developed for simulation of various specific strategies seems promising.

Flow control is closely tied to priority queuing and adaptive routing, although the relationships are not fully understood. Priority queuing delays can be calculated for a variety of priority disciplines, and can be used to reduce average message delay by giving priority to short messages. Priority schemes can also be used for flow and congestion control by limiting entry of or discarding from the network low priority messages, and by increasing the priority of messages which have been a long time in the network.

Effective management of both nodal and destination buffers reduces buffer storage cost. Complete partitioning dedication of buffer space to various outgoing links as well as complete sharing of buffer space among all outgoing links is inefficient in comparison with compromise schemes involving partial sharing and partial dedication. These latter schemes require additional study and evaluation. Actually, buffer costs should be optimized with link flows, link capacities and network topology during network design.

Source-destination controls (SDC's), often called end-to-end controls, are similar to DLC's in that SDC's perform the function of synchronization, error control, and initiation and termination of message transmission. However SDC's operate at the end-to-end multipacket message level whereas DLC's operate on single packets at the node-to-node level. Synchronization involves ordering of packets at the destination prior to delivery to the user. Variations in source-destination packet transmission times resulting from differences in routes, retransmissions, or differing nodal delays virtually prohibit calculation of source-destination message delays or delay distributions. As a result destination reassembly buffer overflow calculations are impossible and must be determined either by simulation or by observation of operating networks. One might expect end-to-end message delay for a particular message type to approximate a Gamma distribution, but evidence to support or refute this conjecture seems unavailable. Even a preliminary study similar to the one on DLC's dealing with calculation and realization of minimum required SDC overhead seems unavailable.

Interest in packet radio networks is increasing, particularly for use in remote areas or with mobile terminals. The broadcast

feature of packet radio creates both opportunities and problems, the latter including potential network overload from generation of multiple copies of a single packet. Special routing algorithms are required which prevent packet proliferation while maintaining network reliability. Quantitative study regarding the design and operation of packet radio networks is in its infancy.

The importance of and motivation for improved understanding of network management issues is provided by knowledge that management information accounts for what is undoubtedly an unnecessarily large fraction of the total network traffic; approximately 90% of the ARPANET traffic is overhead traffic [K2, S6].

Excluded from the present study are considerations regarding protocols for communication between processes [S7, W3, C6]. The problem lies beyond the scope of network design, and involves fundamental considerations regarding procedures and minimal information overhead needed to initiate, terminate, and reliably provide for conversations between processes linked by unreliable communication facilities. Our current understanding of the fundamentals of interprocess communication is minimal.

I-4 Recommendations for Further Study

Listed below are topics which warrant further study:

1. Because any network design must rely to some extent on network component costs and traffic data, improved costs and traffic estimates as well as trends would be useful. Potential user demand and government actions including subsidies will affect traffic and possibly costs; the potential effects of such actions are not fully understood.

2. Efforts should be made to modify existing network design procedures or to develop new procedures which incorporate reliability at the outset as a design constraint. Of particular interest is the quantitative reliability measure proposed in Chapter 2 which defines the reliability (actually unreliability!) as the fraction of traffic unable to reach its destination because of link or node failures. The effects of this constraint on network topology, data link flows, link capacities, and design costs and complexities warrant thorough study.
3. Definition of sensitivity and adaptability measures for incorporation into network design procedures as well as for evaluation of network designs would be useful.
4. A thorough assessment of spread spectrum multiple access for EIS data traffic multiplexing is warranted.
5. Further study of modified contention multiplexing on satellite, radio and other shared channels is needed. Such modifications reduce system performance degradations resulting from "collisions" between two or more message streams. Optimization of system parameters, sensitivity of message delay and throughput to parameter settings and traffic statistics, and implementation cost considerations are of specific interest. Such a study should include delay interactions resulting from queuing and retransmissions caused by "collisions". Also of interest is the approximation of delay-vs-throughput curves by relatively simple expressions to facilitate network design.
6. The need for further study on the topological design of distributed networks is clearly indicated. A major problem is to devise design procedures which generate large networks having acceptable delay, cost, throughput and reliability as well as the capability for message routing which avoids large nodal buffer and traffic overhead costs associated with routing and routing updates. One promising approach involves merging of a distributed network design procedure based on nodal clustering with

recent results for hierarchical routing in large networks.

7. The high overhead associated with data link controls (DLC's) indicates the need for a fundamental study whose objective would be to determine the minimum amount of DLC overhead required under realistic network conditions, and to devise efficient and reliable DLC's which approach this minimum.
8. Additional study on adaptive routing is warranted. Issues of importance include optimal division of centralized and distributed control of routing, development of effective routing algorithms and minimization of routing update overhead.
9. Flow and congestion control is not well understood and requires additional study, which should include priority queuing, adaptive routing and buffer management considerations.
10. Further study is required to devise buffer allocation strategies which compromise complete buffer sharing with total buffer partitioning and subsequent dedication to outgoing links or incoming destination messages. Efforts should be made to incorporate buffer cost optimization with optimization of link flows, link capacities and topology.
11. Source to destination (or end-to-end) controls (SDC's) enable delivery of multipacket messages from source to destination. Fundamental studies are needed to determine the minimum SDC overhead required as well as implementations which approach this minimum. Also needed are estimates end-to-end message delay distributions to facilitate re-assembly buffer optimization and flow control.
12. Packet radio networks offer new challenges, including the need for routing algorithms which avoid packet proliferation while maintaining operational reliability.
13. Related to network design are distributed data base design, interprocess communication and internetworking. Each of these broad subject areas require detailed study to

facilitate implementation of electronic information services.

I-5 References

- A1 N. Abranson and F.F. Kuo, Ed., Computer-Communication Networks. Englewood Cliffs, N.J.: Prentice-Hall, 1973.
- A2 ACM Committee on computers and public policy, D.D. McCracken, Chairman, "A problem-list of issues concerning computers and public policy," *Comm. ACM*, vol. 17, pp. 495-503, Sept. 1974.
- B1 P. Baran, "Broad-band interactive communication services to the home; Part I - potential market demand," *IEEE Trans. Commun.*, vol. COM-23, pp. 5-15.
- B2 G.M. Booth, "The use of distributed data bases in information networks," in [W2], pp. 371-376; also in [C3].
- B3 P. Baran, "Broadband interactive systems to the home; Part II - impasse," *IEEE Trans. Commun.*, vol. COM-23, pp. 166-170, Jan. 1975.
- B4 R.P. Blanc and I.W. Cotton, Computer Networking. New York, N.Y.: IEEE Press, 1976.
- C1 K. Chen, "Cable communications policy issues: an overview," *IEEE Trans. on Systems, Man and Cybernetics*, vol. SMC-6, pp. 727-734, Nov. 1976.
- C2 S.K. Chang, "A model for distributed computer system design," *IEEE Trans. Syst. Man. and Cybern.*, vol. SMC-5, pp. 344-359, May 1976.
- C3 W.W. Chu, Ed., Advances in Computer Communications. Dedham, Mass.: Artech House, 1976.
- C4 V.G. Cerf and R.E. Kahn, "A protocol for packet network inter-communication," *IEEE Trans. Commun.*, vol. COM-22, pp. 637-648, May 1974; also in [C3].
- C5 W.W. Chu, "Optimal file allocation in a multiple computer system," *IEEE Trans. Computers*, vol. C-18, pp. 885-889; also in [A1] and [C3].
- C6 S.D. Crocker, J.F. Meafner, R.M. Metcalfe and J.B. Postel, "Function-oriented protocols for the ARPA computer network," in *AFIPS SJCC Conf. Proceedings*, Montvale, N.J.: AFIPS Press, 1972; also in [C3].

- D1 R.W. Donaldson, "Communications for Text Processing: with Application to Electronic Information Services," Report to the Department of Communications, Ottawa, Canada, Jan. 1977, ch. 4.
- D2 D.A. Dunn and A.J. Lipinski, "Economic considerations in computer-communication systems," in [A1], Ch. 10.
- F1 R. Fajman and J. Borgett, "WYLBUR: An interactive text-editing and remote job entry system," Commun. ACM, vol. 16, pp. 314-322, May 1973.
- F2 N. Foy and W. Helgason, "Europe claims the lead in banking," Datamation, vol. 22, pp. 57-59, July 1976.
- F3 J.D. Foley, "A model of distributed processing in computer networks, with application to satellite graphics," in [P1], pp. 331-335.
- H1 G.D. Hodge, "An electronic mail system -- will it happen?" in [P1], pp. 351-357.
- H2 W.L. Haney, W.C. Brown and J. Brahan, "The computer-aided learning program at the National Research Council of Canada," Int. J. Man-Mach. Studies, vol. 5, pp. 271-288, July 1973.
- H3 R.C. Harkness, "Selected results from a technology assessment of telecommunication-transportation interactions," in Conf. Rec., IEEE Int. Conf. Commun., Philadelphia, Penn., June 1976, pp. (26-1)-(26-7).
- J1 F.J. Jensen, "Centralization or decentralization in banking," Datamation, vol. 22, p. 60, July 1976.
- J2 G.W. Jull, R.W. McCaughern, N.M. Mendenhall, J.R. Storey, A.W. Tassie, and A. Zalatan, "Research Report on Teleconferencing," vol. 1 and vol. 2, CRC Rept. No. 1281-1 and 1281-2, Ottawa, Canada, Jan. 1976.
- K1 R.E. Kahn, "Resource sharing in computer networks," Proc. IEEE., vol. 60, pp. 1397-1407, Nov. 1972; also in [C3].
- K2 L. Kleinrock, Queuing Systems, Vol. 2: Computer Applications. New York: Wiley, 1976.
- K3 T.C.S. Kennedy, "Experience with a mini-computer based hospital administration system," Int. J. Man-Mach. Studies, vol. 5, pp. 237-250, March 1973.
- K4 T.C.S. Kennedy, "The design of interactive procedures for man-machine communications," Int. J. Man-Mach. Studies, vol. 6, pp. 309-334, May 1974.
- K5 P.T.F. Kelly, "An overview of recent developments in common user data communication networks," in [V2], pp. 5-10.

- K6 R.E. Kahn, "Organization of resources into a packet radio network," IEEE Trans. Commun., vol. COM-25, pp. 169-178, Jan. 1977.
- K7 H. Katzan, Jr., Computer Data Base Management and Data Base Technology. New York, N.Y.: Van Nostrand, 1975.
- L1 C.M. Laucht, "Electronic mail for the Canadian environment," in Conf. Rec., IEEE Int. Conf. Commun., Philadelphia, Penn., 1976, pp. (15-3)-(15-4).
- L2 H. M. Lipinski, and R.H. Miller, "FORUM, a computer-assisted communications, medium," in [P1], pp. 143-147.
- M1 J. Martin and A. Norman. The Computerized Society. Englewood Cliffs, N.J.: Prentice-Hall, 1970.
- M2 W.J. Miller, "Technology assessment for electronic message handling," in Conf. Rec., IEEE Int. Conf. Commun., Philadelphia, Penn., 1976, pp. (15-10)-(15-12).
- M3 J. Martin, Design of Man-Computer Dialogues. Englewood Cliffs, N.J.: Prentice-Hall, 1973.
- M4 J. Martin, Principles of Data-Base Management. Englewood Cliffs, N.J.: Prentice-Hall, 1976.
- M5 J. Martin, Computer Data-Base Organization. Englewood Cliffs, N.J.: Prentice-Hall, 1975.
- M6 W. H. Melody, "Relations between public policy issues and economics of scale," IEEE Trans. on Systems, Man, and Cybern., vol. SMC-5, pp. 14-22, Jan. 1975.
- M7 J. Martin, Systems Analysis for Data Transmission. Englewood Cliffs, N.J.: Prentice-Hall, 1972.
- N1 J.M. Nilles, F.R. Carlson, P. Gray and G. Hanneman, "Telecommuting - an alternative to urban transportation congestion," IEEE Trans. Systems, Man, and Cybern., vol. SMC-6, pp. 77-85, Feb. 1976.
- O1 S.S. Oren, "A mathematical theory of man-machine document assembly," IEEE Trans. on Systems, Man, Cybern., vol. SMC-5, pp. 520-527, Sept. 1975.
- O2 S.S. Oren, "A mathematical theory of man-machine text editing," IEEE Trans. on Systems, Man, Cybern., vol. SMC-4, pp. 258-267, May 1974.
- P1 Proceedings of the Second International Conference on Computer Communications. Stockholm, Sweden, August 1974.

- P2 G.E. Passant, "Operational impact on real time computing in trustee savings banks," in [P1], pp. 83-85.
- P3 H.A. Peele and E.M. Riseman, "The four faces of HAL: a framework for using artificial intelligence techniques in computer-assisted instruction," IEEE Trans. on Systems, Man and Cybern., vol. SMC-5, pp. 375-380, May 1975.
- P4 D.F. Parkhill, "Society and computer communication policy," in [V2], pp. 11-17.
- P5 R.R. Panko, R.W. Hough, and R. Pye, "Telecommunications for office decentralization: apparent needs and investment requirements," in Conf. Rec., IEEE Int. Conf. Commun., Philadelphia, Penn., June 1976, pp. (26-8)-(26-13).
- P6 R. Pye and P.I. Weintraub, "Social and organizational implications of decentralization," in Conf. Rec., IEEE Int. Conf. Commun., Philadelphia, Penn., June 1976, pp. (26-14)-(26-19).
- P7 R.L. Pickholtz, Ed., "Special Issue on Computer Communications," IEEE Trans. Commun., vol. COM-25, Jan. 1977.
- R1 L.G. Roberts, "Data by the packet," IEEE Spectrum, vol. 11, pp. 46-51, Feb. 1974; also in [C3].
- R2 M.A. Robbins, "Error objective in the electronic mail system," in Conf. Rec., IEEE Int. Conf. Commun., Philadelphia, Penn., 1976, pp. (15-7)-(15-9).
- R3 M.L. Rockhoff, "An overview of some technological/health-care system implications of seven exploratory broad-band communication experiments," IEEE Trans. on Commun., vol. COM-23, pp. 20-30, Jan. 1975.
- R4 L.G. Roberts, "Extensions of packet communication technology to a hand held personal terminal," in AFIPS Conf. Proceedings, SJCC, vol. 40, Montvale, N.J.: AFIPS Press, 1972, pp. 295-298; also in [C3].
- R5 L.G. Roberts, "International interconnection of public packet networks," in [V2], pp. 239-245.
- S1 M. Schwartz, R.R. Boorstyn and R.L. Pickholtz, "Terminal-oriented computer-communication networks," Proc. IEEE, vol. 60, pp. 1408-1423, Nov. 1972; also in [C3].
- S2 B. Skoldborg, "Real time banking system as an application," in [P1], pp. 71-82.
- S3 M.E. Silverstein, S. Rosenberg, and M.A. Cremer, "Medical diagnostic terminals: their nature and description of an archetypal labour saving system," in [P1], pp. 53-58.

- S4 M.E. Silverstein, "The wired medical community: an emerging reality," in [P1], pp. 41-45.
- S5 R.K. Summit and O. Firschein, "On-line reference searching," IEEE Spectrum, vol. 12, pp. 68-71, Oct. 1975.
- S6 M. Schwartz, Computer Communication Network Design and Analysis. Englewood Cliffs, N.J.: Prentice-Hall, 1977.
- S7 C.A. Sunshine, "Interprocess Communication Protocols for Computer Networks," Stanford Univ. Digital Systems Laboratory Tech. Rept. 105, Stanford University, Stanford, Calif., Dec. 1975.
- S8 T.D. Sterling, "Guidelines for humanizing computerized information systems: a report from Stanley House," Commun. ACM, vol. 17, pp. 609-613, Nov. 1974.
- T1 J.M. Taplin and M.P. Beere, "The economics of new information networks," IEEE Trans. on Systems, Man, and Cybern., vol. SMC-5, pp. 40-43, Jan. 1975.
- V1 J. Vallee, R. Johansen, H. Lipinski, and T. Wilson, "Pragmatics and dynamics of computer conferencing: A summary of findings from the FORUM project," in [V2], pp. 208-213.
- V2 J.J. Verma, Ed., Proceedings of the Third International Conference on Computer Communication, Toronto, Canada, August 1976.
- V3 H. Von Baeyer, "The quest for public policies in computer/communications - Canadian approaches," in [P1], pp. 19-23.
- W1 R.L. Williams, "Mailgram -- an electronic mail service," in Conf. Rec., IEEE Int. Conf. Commun., Philadelphia, Penn., 1976, pp. (15-5)-(15-6).
- W2 S. Winkler, Ed., Computer Communication, Impacts and Implications: Proceedings of the First International Conference on Computer Communication. Washington, D.C., October, 1972.
- W3 D.C. Walden, "A system for interprocess communication in a resource sharing computer network," Commun. ACM, vol. 15, pp. 221-230, Apr. 1972; also in [C3].

II FORMULATION OF THE NETWORK DESIGN PROBLEM

II-1 Representative Electronic Information Service (EIS) Applications

Formulation of the EIS network design problem is facilitated by considering briefly some applications to be served by the network. As noted in Chapter 1, typical applications include word processing [M1, O1, O2, F1, N1], electronic mail [H1, L1, M2, W1, R1], banking and electronic funds transfer [J1, F2, S1, P1, M1, B1] and point-of-sale systems [M3, M1].

Word processing includes the assembly of documents from stored text segments, text editing, spelling correction and final document preparation. Although specific word processing system configurations vary widely, a system would normally include a keyboard, a CRT display, digital storage, logic for searching, retrieving, updating and editing stored information, and a hardcopier [O1, O2]. A typical system user would be a lawyer who must prepare affidavits and other documents comprising standard paragraphs or minor variations thereof. For example, a mortgage would be prepared by manually typing the name, address and occupation of each of the parties. Appropriate paragraphs would then be retrieved from storage, edited, and assembled for proofreading and subsequent hardcopying. Special non-standard clauses could be inserted manually prior to final hardcopy production.

Several individuals could work simultaneously on the preparation of a single document provided each worked on a different segment. The individuals could be in different geographical locations [N2, P2, P3] provided each had access via a terminal to a

central storage and processing facility. Alternatively, several remote users could prepare their own individual documents, with the central location being used for storage of standard text segments, processing, and storage of software for text editing and assembly.

In its initial configuration an electronic mail system might consist of electronic mail centres (EMC's) located at various population centres throughout the country. Paper mail would be brought to an EMC for conversion via facsimilie scanners or optical character readers to an electronic format suitable for transmission over a data link to a recipient EMC. "Mail" stored digitally on magnetic tape would also be brought to the EMC for transmission. "Mail" could be transmitted via data lines from the senders' premises to the EMC. Recipient EMC's would store received "mail" electronically for delivery. For recipient users having appropriate facilities, distribution would involve delivery of either the actual magnetic tapes to the recipient's premises or transmission of electronic signals to the recipient user's electronic storage facilities. For other recipient users, electronic mail would be converted at the recipient EMC to hardcopy format for conventional hand delivery.

An electronic mail facility would undoubtedly foster new kinds of message communication. For example, voice gram [K1] has been proposed; the senders voice would be converted to digital format, transmitted as "mail", and later reconstructed. Acoustic transducers would be required at the sending and receiving end of the system. A voice-gram system would obviate the necessity for preparing and sending a letter if a called party could not be reached, a situation which occurs, allegedly, with a probability

of at least 0.9 on any given attempt [K1]. Voice-gram might be attractive for use with mobile terminals, since many vehicles already possess the required transducers, modems and antennas.

Hand-held terminals have also been proposed [R2] for use as terminal-interfaces to electronic information systems. Such terminals would complement and possibly replace existing paging systems and might be useable as terminals in automobiles and other vehicles.

Financial institutions are currently in various stages of automation. In many banks a teller can visually examine a daily computer printout detailing an account's current status. In some banks the information is available within a few seconds via a keyboard-display which communicates with a CPU which stores, retrieves, and updates each account. The number of banks with electronic cash dispensers continues to increase.

Closely related to banking and electronic funds transfer (EFT) are point-of-sale (POS) systems [M1, M3]. Terminals at cashier stations communicate with a CPU which uses information received to debit or credit customer accounts, update inventories, and record sales commissions.

It is not difficult to visualize a merging by stages of electronic banking and POS systems. Accounts at retail outlets would communicate with or be merged with bank accounts. Many kinds of financial transactions would then be executed electronically, with actual cash payments eventually being reserved for payment of the small corner grocer, delivery boys and parking meters.

II-2 Network Message Statistics

Perhaps the most significant fact concerning message statistics for EIS applications is that actual data is very scarce. As indicated in the previous section, most implementations are either recent, in progress or pending. Obtaining actual data is expensive and time-consuming, and in many cases virtually impossible without causing some network disruption or violation of proprietary constraints. It is often argued that changes inherent in a growth industry would soon render any existing statistics obsolete.

A model for the average traffic r_{ij} from location i to location j , proposed by Zipf [Z1] and subsequently used by Kleinrock [K2] defines

$$r_{ij} = P_i P_j / d_{ij} \quad (2-1)$$

where P_i denotes the population of the region containing location i and d_{ij} denotes the geographical distance between locations i and j . Use of (2-1) requires definition of regions, which presents immediate difficulties. Distance d_{ij} would often require modification to account for political, cultural and geographic boundaries. Nonetheless, the model has intuitive appeal.

Concerning the distributions of message lengths, times between message characters and times between messages in interactive computer-communication applications, some data are available [D1, F3]. One study [F3] deals with long holding times of between 15 and 30 minutes while another [D1] deals with short holding times of between one and two minutes. Long holding times are typical of business and scientific applications requiring extensive computation as well as transfers of long data files. Short hold

times occur in enquiry-response systems involving on-line banking, credit checking and production control.

Both studies show the channel to be idle much of the time. In long holding time situations, the user is active approximately 5% of the time and the response computer approximately 30% of the time [F3]. In short hold situations the activity times are 15% and 35% for the user and computer, respectively [D1]. This partial channel utilization is also present in voice systems, and several proposals have been made for transmitting data during these idle periods [A2, R3].

In dealing with message arrivals, it is of the greatest analytical convenience to assume that messages arrive independently of each other, in which case the probability of exactly n messages arriving during a time interval τ is [P4]:

$$P_n(\tau) = \frac{(k\tau)^n}{n!} \exp(-k\tau) \quad (2-2)$$

where k is the average message arrival rate. Traffic statistics [F3] support the Poisson assumption which implies exponential inter-arrival times.

In both studies [D1, F3] cited earlier the times between characters or bursts of characters is well approximated by a Gamma distribution whose probability density function $f_x(U)$ is as follows [P4]:

$$f_x(U) = \begin{cases} \frac{c^{b+1}}{\Gamma(b+1)} U^b \exp(-cU) & U \geq 0 \\ 0 & U < 0 \end{cases} \quad (2-3)$$

where $\Gamma(\cdot)$ is the Gamma function, and b and c are non-negative constants which define the distribution. The first and second

moments of the distribution are, respectively,

$$E(x) = (b+1) / c \quad (2-4)$$

$$E(x^2) = (b+1)(b+2) / c^2 \quad (2-5)$$

where $E[.]$ denotes expected value.

The ratio $\phi = E(x^2) / E^2(x)$ is also of interest in evaluation queuing delays (see Chapter 3). For the Gamma distribution

$$\phi = (b+2)/(b+1) \quad (2-6)$$

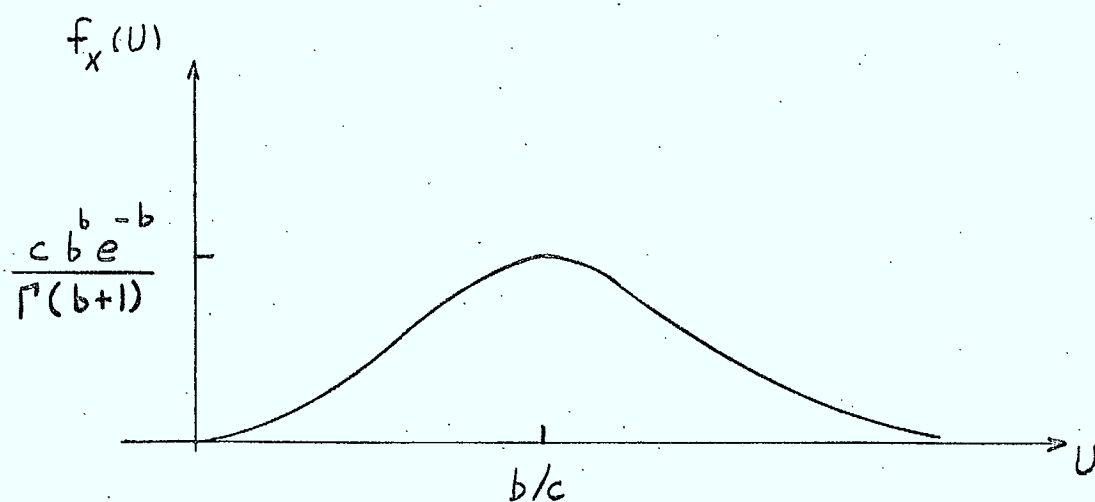
Fig. 2-1 shows both $f_x(U)$ and $\phi(b)$. It is seen that $f_x(U)$ has a single peak at $U=b/c$ and that $1 \leq \phi \leq 2$. If $b \rightarrow 0$, ϕ is maximized and the resulting distribution is exponential with mean value $1/c$. If $b \rightarrow \infty$, $c \rightarrow \infty$ and b/c remains constant, then the constant (impulse) distribution of length b/c results. The Erlengian distribution is a special case of the Gamma distribution and results when b is restricted to integer values $[M4]$, in which case $\Gamma(n+1)=n!$ [P4].

Message length distributions are often approximated by the geometric distribution

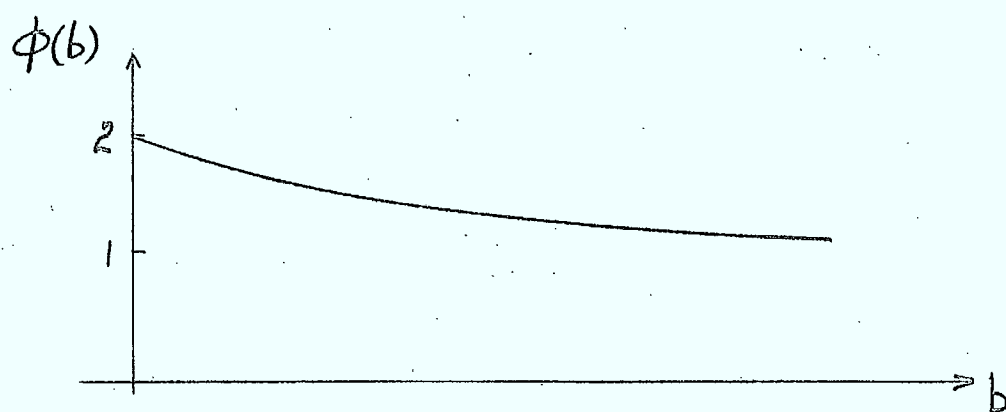
$$g(l) = (1-p)^{l-1} p \quad (2-7)$$

where integer l is the discrete message length and parameter p defines the average message length which equals p^{-1} [S2]. For long messages the continuous exponential distribution with density function

$$f_x(U) = \begin{cases} ce^{-cU} & U \geq 0 \\ 0 & U < 0 \end{cases} \quad (2-8)$$



(a)



(b)

Fig. 2-1 Gamma distribution (a) Density function $f_x(u)$ (b) Moment ratio.

with mean length c^{-1} is used to closely approximate the geometric distribution. The means of the two distributions are equal if $c=p$. The study by Fuchs and Jackson [F3] indicates that the geometric distribution is a reasonable fit to most of their message length data.

The fact that EIS applications include both file transfers involving long messages (typically 10^4 to 10^7 bits [K3] and interactive conversations involving short messages (typically 20 to 600 bits [K3]) suggest a bimodal message length distribution. No single unimodal message length distribution in the midrange seems to offer a suitable fit to the two types of data [K3, R4, K4]. From these discussions one reaches the following conclusions:

1. EIS messages show enormous variations in length.
2. Actual message statistics are sparse.
3. Message statistics and traffic estimates may be subject to rather rapid and continuing change.
4. Network design procedures should recognize items 1-3 above.

II-3 Network Costs and Performance Criteria

An important and often underemphasized aspect of system analysis, design and evaluation is the selection of performance criteria, which should be meaningful, quantitative, measurable and analytically tractable.

Appropriate EIS network performance criteria include message delay, message throughput and network reliability. Each of these is briefly discussed below following discussions of network costs.

Costs include hardware items such as data channels,

concentrators, multiplexors, buffer storage, and CPU's as well as software for network operation and management.

Construction costs for terrestrial data channels increase stepwise with capacity and linearly with distance [C2, D2, D3, D4]. Rental costs increase stepwise with both capacity and distance [D2, D3, D4, G1]. Representative cost schedules show that costs increase more slowly than capacity and for rental costs, more slowly than distance. Cost-vs-distance relationships are often further complicated by rate variations resulting from provincial or state jurisdictions traversed by data links.

The 11 percent per annum decrease in terrestrial cost in the USA since 1960 [R5] is due to improved solid state technology, increased subscriber volume, and increased modem data rates as explained in Section 3-2. Further real (excluding inflation considerations) cost decreases of 50 percent per decade (7 percent per annum) seem likely during the next twenty years [D2, R3]. The technological breakthrough which might precipitate a further large cost decrease is the realization of fibre optic channels [C3].

Satellite data channel costs have decreased at a rate of 40 percent per annum during the past few years [R5]. It seems unlikely that this rate of decrease will be sustained as the technology matures.

Concentrator and multiplexor costs increase with data handling capacity, but at a rate which decreases as the capacity increases [F4, T1, C2]. However, when these are used as components for nodal switches in distributed networks, the cost seems to increase linearly with the number of terminations [R3]. Improved microprocessor and LSI technology will undoubtedly result in sub-

stantial cost decreases in concentrator, multiplexor and switch costs in the coming years.

Buffer storage costs include an overhead cost plus a factor which tends to increase linearly with the capacity of any one type of storage medium. A variety of media including semi-conductor memory, core, disc, drum and tape are available. The slower the memory access speed, the lower the cost per bit. Large decreases in core and disc costs seem unlikely [D2]. Magnetic tape costs may decrease if signal design and detection techniques used on bandlimited data transmission channels (see Section 3-2) can be modified to increase data packing density [M5]. Increasing the packing density would also decrease the access time. Drastic cost reductions in semi-conductor memories are not expected; however, forecasting here is difficult [H2]. Optical, magnetic bubble and thin film storage methods are emerging technologies which offer some likelihood of further cost reductions and fast access rates [C4, H3, B2]. Accurate predictions of costs and availability dates is extremely difficult.

CPU costs have decreased by a factor of 10 over the past five years, and will likely continue to decrease substantially, [D2] in part because of improvements in LSI and microprocessor technology.

Software costs have increased over time to the point where these often constitute the major cost component of a network. The following comments provide some appreciation for software cost trends [D2]:

1. The programming cost per phrase remains constant over time.

2. The cost per phrase increases with program size; if a 1000 phrase program cost \$5.00 per phrase a 10,000 phrase program might cost \$10.00 per phrase.
3. The cost of writing a program for a specific task decreases at an annual rate of 25 percent.

As network management protocols become standardized, one would expect software costs to decrease by virtue of item 3 above. Another trend is for software design to become a formal engineering discipline with associated cost controls. However, programmers often work as artisans who do not always conform to a disciplined mould.

The cost of a network is the sum cost of its component parts. The fact that various components exhibit different rates of cost change poses difficulties for network designers, who must attempt to anticipate costs in effect when the network is to be in operation, and who must consider the sensitivity of the network design to cost changes.

User terminal and local file storage costs are not discussed here, since these are not normally regarded as network costs. These costs, as well as potential demand for EIS services and effects of government policies on costs and demand are considered in an earlier report by the author [D5].

Message delay is of utmost concern in on-line applications involving interactive dialogue; included here is on-line computing, banking, point-of-sale and word processing. Message delay is usually defined as the time between the offering of the last character of a message to the data link and the last character's arrival at the destination. Network delays occur because of the

finite rate at which data is transmitted over a channel, queues which form at network nodes, nodal processing delays which include assignment of messages to appropriate outgoing links, modem turn-around times, and retransmissions necessitated by data link errors, network component failures, or contention multiplexing collisions. Delays of from 2 to 3 sec. with a 1 to 2 sec. variance is normally considered reasonable and acceptable [M6, R6]. Response behaviour expected by on-line users of electronic information services is similar to that expected of other humans [M6, R6, K5, D6, H4, F5].

Delay usually implies an average over all data in the network, often at the busiest time of day. The delay as seen by an individual user can vary considerably from the average; similarly, an individual user's variance can be different from that for the network. To include individual user delays as explicit performance criteria in design would be computationally prohibitive and probably ineffective because of the unreliability of individual user statistics. Alternatives to maintaining satisfactory performance for individual users include priorities for a fee, congestion control, and adaptive routing, as explained in Chapter 5.

Data throughput is another performance measure of interest. Throughput is the total data traffic from source to destination per unit time.

Reliability is of interest for obvious reasons. The difficulty in deciding upon an appropriate definition for reliability is reflected in the number of alternative definitions which have been proposed [W2, T2, V1, O3, L2, H5, H6, K6, F6, F7, F8, D7, C5, A3, A4].

Reliability criteria are often characterized as either

deterministic or probabilistic [W2]. Deterministic criteria are based on measures indicating the number of network links or nodes that can fail without disrupting network operation [W2, F8, A3, T2]. Deterministic criteria were originally based on the presumed existence of a human adversary with knowledge of the network structure and therefore indicated how difficult it would be for such an adversary to completely disrupt communication. Deterministic criteria are based on network topology and graph theoretic concepts and do not normally provide quantitative measures of the degree of delay or throughput degradation caused by non-uniform link traffic.

Probabilistic reliability criteria [W2, F7, K6, D7, H5] are more appropriate for EIS applications because they tend to indicate the degree to which a network fails to perform as intended. Probabilistic measures include the probability that two network nodes selected at random will not be able to communicate, or the probability that all operating nodes will be converted to the network by at least one link, or the expected fraction of communicating node pairs. Some probabilistic criteria are rather strongly dependent on topology. For example, when nodes are virtually 100 percent reliable and all links are equally unreliable and fail independently of each other, one can enumerate the number of network states which result in a disconnected network, calculate the probability of each state, and sum these to obtain the probability of a disconnected network.

There seems to be no network design procedure which formally incorporates reliability as a design constraint. Some procedures

include ad-hoc constraints such as "all nodes must terminate at least two links for reliability purposes." Reliability is often assessed following completion of the design process. In the next section we propose a formal reliability constraint for inclusion at the outset of the network design process.

Related to but distinct from reliability are sensitivity and adaptability. Sensitivity relates to the amounts by which various performance criteria change in response to changes in design information such as data link costs, traffic or link failure probabilities. Sensitivity is particularly important when the original design information changes often and by large amounts. Adaptability is closely related to sensitivity, and indicates the ease with which an original network design can actually be modified in response to changes in design data.

The EIS industry is subject to changing rate structures and costs of technology, and to growth industry traffic changes. Because the amounts and timing of such changes are not easily predicted, sensitivity and adaptability considerations are important. Like reliability, they are not normally incorporated at the outset in the network design procedure and are often ignored altogether until network operation and management is considered. We would propose that network designs be subjected to sensitivity studies, and be evaluated as to their adaptability.

II-4 Network Design Problem Statement

In analysis and design of large systems problem formulation is of utmost importance. Problem formulation involves specification of a system model which is sufficiently accurate to represent the

salient aspects of system behaviour, sufficiently simple to permit meaningful analysis and system evaluation, and sufficiently flexible to accommodate modifications. Problem formulation and mathematical modelling is normally an iterative procedure in which the model is improved as more is learned from studies of earlier models and from experiments suggested by these models [S3].

In designing large, complex systems, one does not usually insist on optimum designs, particularly when the design data is sparse and subject to change. Efforts are devoted instead to avoiding crippling non-optimalities and to obtaining good solutions which remain good or are easily modified if the design data changes.

Fig. 2-2 shows an electronic information network consisting of N external nodes at which are located terminals, computers, computer peripherals or memory storage banks, and internal (backbone) nodes which store and forward messages whose final destination is normally an external node. The network consists of M links. Some links carry traffic in one direction only, with the result that traffic can flow over a pair of such links in both directions along a path between two nodes to provide full-duplex (FDX) operation. Some networks or portions of networks use half-duplex (HDX) links, in which case traffic can flow in either one direction or the other but not both at the same time. In the HDX case the total traffic λ includes the two one-direction traffic components. Some lines (normally HDX) support terminals in multidrop fashion, in which case the number of terminals operating on the line at any one time is restricted.

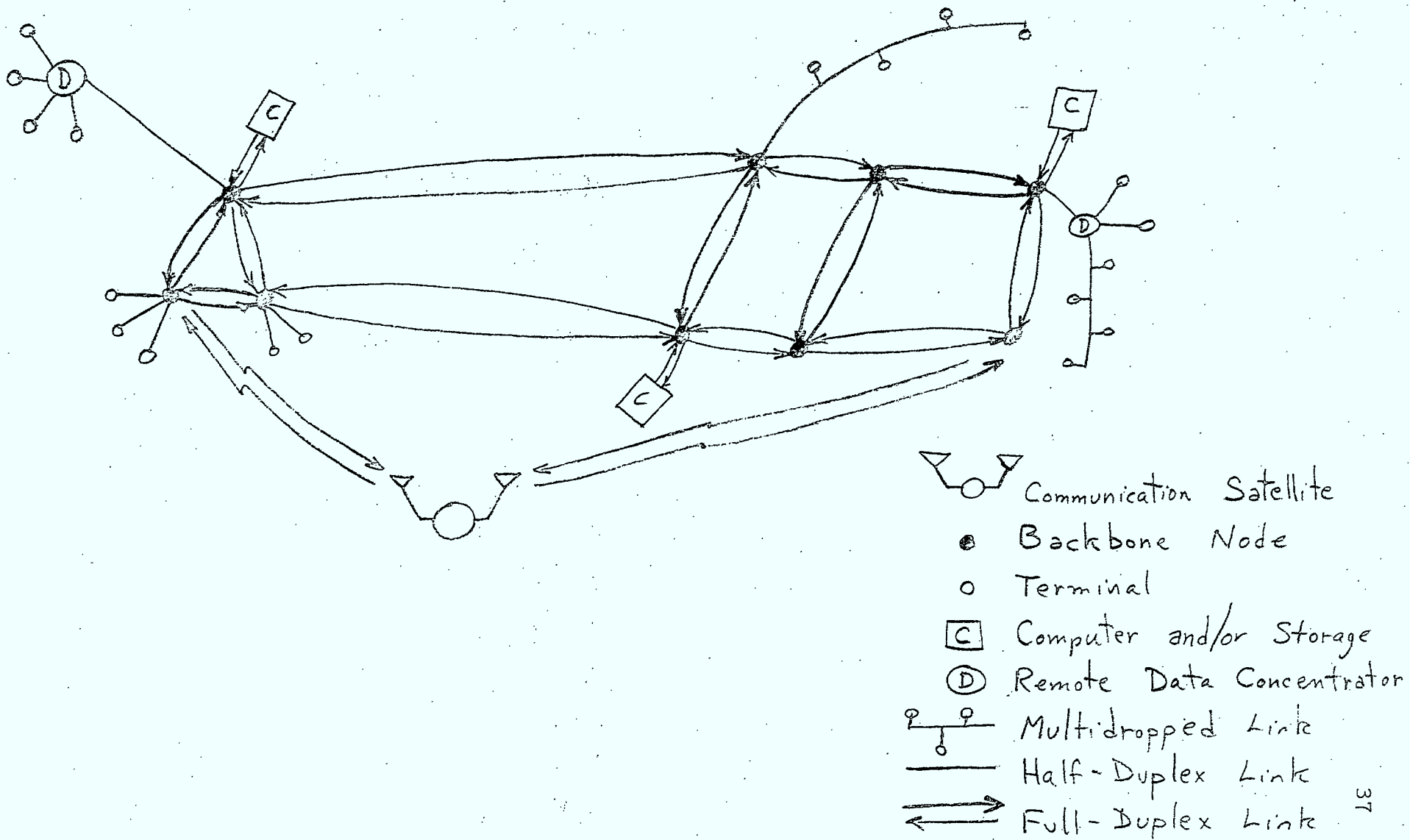


Fig. 2-2 Typical electronic information network.

The data usually specified at the outset is the traffic matrix $[r]$, defined in Section 2-2. Either average message delay or network cost is optimized subject to a constraint on the other quantity. As noted earlier, reliability is not normally incorporated at the outset as a formal design constraint, but is considered later in an ad-hoc way.

The design variables are many and include data link capacities, data link traffic flow, message routing which may be either fixed or adaptive, nodal buffer size and organization, network topology, data channel multiplexing or polling techniques and congestion control strategies.

Define total network cost, average delay, unreliability and throughput, respectively, as D , T , U and γ . It follows that

$$\gamma = \sum_{i=1}^N \sum_{j=1}^N r_{ij} \quad (2-9)$$

In (2-9) γ denotes the level of network traffic; a k -fold increase in γ implies a k -fold increase in all matrix entries r_{ij} .

If the cost is the performance measure then for a given set of network variables

$$D = D(T, U, \gamma) \quad (2-10)$$

One expects an increase in T , an increase in U or a decrease in γ to reduce D . On the basis of the discussions in Section 2-1, the cost per bit D/γ should decrease with an increase in γ . Such behaviour has been observed [D3].

With delay T as the performance measure

$$T = T(D, U, \gamma) \quad (2-11)$$

An increase in D , which implies high capacity data links and/or more data links, as well as a decrease in U or γ would normally reduce T .

If T_i and λ_i denote, respectively, the average delay and message arrival rate on link i , then [K4]

$$T = \sum_{i=1}^M (\lambda_i / \gamma) T_i \quad (2-12)$$

Note that (2-12) does not require that any assumptions regarding the independence of message arrivals at nodes. However, for analytical purposes evaluation of the nodal delays T_i does require the assumption that messages arrive independently at each node. The calculation of T_i for queuing and various channel multiplexing techniques is considered in Chapter 3.

It is important to realize that T in (2-12) does not represent average delay for messages which are broken into smaller messages or packets for transmission by the network, but for the packets themselves. Delay for the original message is always of primary interest, but difficult to calculate since the individual packets experience different delays [C6, S4]. Small values of T in (2-12) for message packets does imply reduced delay for original messages [S4] (see Section 5-8).

Simulation studies indicate that average delays calculated using the independence assumption are reasonably close to observed delays [K2]. Efforts have been made to include the effects of dependencies in message delays [K7, R7, K8]; however these approaches are either topologically restrictive or analytically unwieldy. One could argue that because of the imprecision and continual changes

in data used for design, the independence assumption will likely yield networks having performance characteristics comparable to those which might result if more accurate delay calculations were employed. From such arguments it follows that effort should not be spent on the difficult task of improving our understanding of queuing and message flow dependencies, but rather on the improved management of networks designed using imprecise data. Others would argue that improved management itself requires better knowledge of message statistics and dependencies.

As noted in Section 2-3, network cost is the sum total of all component costs. Costs are sometimes assigned solely to data links for convenience, in which case nodal and operating costs are divided and added to link costs.

It remains to define unreliability U . We propose the following definition; with n as the total number of network nodes:

$$U = \frac{1}{Y} \left[\sum_{i=1}^M p_i \lambda_i + \sum_{j=1}^n U_j \left(\sum_{\text{node } j} \lambda_i \right) \right] \quad (2-13)$$

where p_i is the probability of failure of link i , U_j the probability of failure of node j and $\sum_{\text{node } j} \lambda_i$ denotes the sum total of traffic passing through node i . The definition in (2-13) incorporates both link and node failures, and implicitly assumes the failures are statistically independent. This latter assumption is not always realistic; for example, in radio networks environmental disturbances may cause all nodes and links in a region to fail simultaneously [B3]. However, the statistical independence assumption is reasonable in many situations, and is normally employed in reliability calculations [W2, B3].

The reliability measure in (2-13) represents the fraction of traffic unable to reach its destination because of link and node failures, and indirectly incorporates most other unreliability measures including such as time between failures and mean time to repair [C5, D7]. Analysis using the above definition of U is often simplified if node failures are incorporated with link failures in which case

$$U = \frac{1}{Y} \sum_{i=1}^M \lambda_i (p_i + \sum_{j \in \lambda_i} U_j) \quad (2-14)$$

where $\sum_{j \in \lambda_i}$ includes those nodes which terminate link i . Equation (2-14) assumes that failure of a node involves total failure in the sense that all communication through the node is disabled. Incorporation of node failures with link failures has been proposed by Aggarwal [A4].

The definition in (2-13) or (2-14) involves link flows λ_i as well as network topology which affects both p_i and U_i . It is not unreasonable to assume that p_i is proportional to the length of link i . The more links which terminate on node j , and the larger $\sum_{\text{node } j} \lambda_i$ the larger will be the amount of buffer storage and processing capacity at node j ; hence the larger U_j .

The actual link capacities C_i may affect U through the dependence of p_i on C_i . For example, p_i for cable of capacity C_i may be lower than that for the same length of microwave channel of capacity xC_i where $x \gg 1$, even though p_i is independent of C_i for any one type of physical channel.

As with delay T in (2-12), U in (2-13) and (2-14) applies

not directly to the original messages which may have been decomposed into smaller packets, but to the packets themselves. The relationship of packet to message reliability is considered in Section 5-8.

We noted earlier that the variance σ_T^2 of the average delay T was of interest. If the delays T_i in (2-12) are statistically independent, as is usually assumed, then

$$\sigma_T^2 = \sum_{i=1}^M (\lambda_i / \gamma) \sigma_{T_i}^2 \quad (2-15)$$

where $\sigma_{T_i}^2$ is the variance of nodal average delay T_i . Section 3-3 considers the calculation of $\sigma_{T_i}^2$.

The network design problem involves optimization of one performance variable with respect to network variables, with constraints on the remaining variables. Traffic level γ is normally specified at the outset. Either cost or delay is normally selected as the variable to be optimized, with the other constrained. We would also advocate a constraint on the reliability. In fact, it would be of interest to examine the effect of optimizing U , while constraining T and D . What type of network would result? We don't know.

Some assessment of the network's sensitivity and adaptability is obtained by varying design data such as traffic r_{ij} and component costs, and observing the effects on D , T and U .

II-5 Circuit, Message and Packet Switching

During the early years of telephony digital circuits and the associated technology with its inherent flexibility was non-existent. Virtually all switching was performed manually. Many

parties shared a single local line, and a common form of entertainment was to listen to a neighbour's conversations. The above type of system used circuit (or line) switching, whereby the conversing parties held physical circuit connections until completion of their call. While circuit switching did (and still does) provide for a dedicated connection between two users, the time taken to establish the connection, the rather high probability of a busy signal on one or more of the component links and the unavailability of the circuit to other users during periods of silence (which constitutes at least 50 percent of the call time [R3] makes circuit switching inefficient for EIS applications.

Telegraph systems, many of which originated with the railroads, were and are used to transmit coded messages. In early systems messages arriving at a switching centre were received and stored on punched paper tape. The tape holding the incoming message was placed in appropriate outgoing tape readers for transmission to the next node. The message was transmitted even if subsequent links in the total communication path between source and destination were temporarily unavailable. The price of this convenience includes message storage facilities at each node, as well as variable nodal delays of unknown magnitudes.

Packet switching involves decomposition of a message into fixed-length packets and transmission of these individually, possibly over different routes, to the message destination where message reassembly occurs. Packet switching is particularly attractive when many intermediate nodes lie between the message's origin and destination. As soon as the first packet of a message

is received by the first node in the chain, it can be forwarded while the next packet can be sent from the origin to the first node. If there are more intermediate nodes than packets in a given message the entire message can be moving towards its destination with each packet on a different link. This pipelining effect reduces the source-to-destination message delay $[K_4, S_2]$. For example, if a message consists of y packets and must traverse y links, the time required for transmission will be $1/y$ times that required using message switching. The above statement ignores queuing delays, packet overheads, nodal processing and packet re-transmissions. Packetizing messages can reduce the potentially large nodal storage required to handle very long messages and can provide for additional flexibility regarding message transmission priorities. Thus, single message packets can be interleaved with a sequence of packets from a long message which may have no urgency regarding delivery time.

The above discussions as well as analyses described in Chapter 3 motivate use of either message or packet switching when traffic is bursty, in which case a high bandwidth low duty-cycle channel is needed. Long file transfers, on the other hand, would seem to favour circuit switching. When traffic consists of both interactive conversations and file transfers, packet switching is indicated, for reasons stated in the previous paragraph. Packet switching involves problems additional to those inherent in message switching, which problems include packets arriving at the destination out of sequence, in duplicate or not at all $[S_4, K_4, R_4]$.

The decision as to which form of switching to use is not an

easy one. Some studies have been conducted in order to compare switching techniques [C7, I1, M7], but a comprehensive treatment seems unavailable. Because we do not yet fully understand all the issues involved in the design of packet switched networks, particularly the issues involving network management and protocols, a definitive comprehensive comparison of switching methods seems unlikely at present. However, enough is known concerning the advantages, design and operation of packet switched networks to warrant their continued use and study.

The important issue of packet length poses difficulties [S4, R4]. If H is the packet overhead, including bits for packet synchronization, addressing, error detection and L the packet's maximum length, the following facts are relevant [R4]:

1. The ratio of useful information to overhead $(L-H)/H$ and hence the network throughput γ increases as L increases.
2. The larger L , the larger the probability of a packet error and subsequent retransmission, with an accompanying reduction in throughput if the error rate is too high. Retransmission strategies are considered in Sections 5-3 and 5-8.
3. Increasing L increases network delay for short packets which must wait in a queue behind one or more long packets, some or all of which may be a part of a long message. This particular problem can be obviated by giving short packets priority over long packets. However this queuing discipline discriminates against long messages. Priority disciplines are considered in Section 5-4.
4. It is desirable to make L sufficiently large that most message lengths will not exceed L . In this case, over-

heads and occurrences of out-of-sequence packet arrivals at the destination are minimized.

5. The larger the number of nodes between source and destination, the shorter the desired value of L in view of delay reductions via the pipeline effect.

Issues affecting packet length considerations appear in Sections 3-8, 5-2, 5-3, 5-4 and 5-8.

The few quantitative results available suggest that network performance does not change rapidly as the packet length varies about the apparent optimum value [R4]. The need for more information regarding selection of packet length, particularly as the selection relates to network management and protocols, as well as nodal buffer size and management, would be useful [R4, K4, C6, S4].

The routing of EIS messages over two or more networks which may use different switching mechanisms requires considerations involving interconnection or integration of various types of networks [F9, J2, C8, S4]. Interest continues to grow regarding this matter.

II-6 References

- A1 N. Abramson and F.F. Kuo, Eds., Computer-Communication Networks. Englewood Cliffs, N.J.: Prentice-Hall, 1973.
- A2 K. Amano and C. Ota, "Digital TASI system in PCM transmission," in Conf. Rec., IEEE Int. Conf. Commun., June, 1969.
- A3 R.N. Allan, R. Billington and M.F. De Oliveira, "An efficient algorithm for deducing the minimal cuts and reliability indices of a general network configuration," IEEE Trans. Rel., vol. R-25, pp. 226-233, Oct. 1976.
- A4 K.K. Aggarwal, J.S. Gupta, and K.B. Mistva, "A simple method for reliability evaluation of a communication system," IEEE Trans. Commun., vol. COM-23, pp. 563-566, May, 1975.
- B1 P. Baran, "Broad-band interactive communication services to the home; Part I - potential market demand," IEEE Trans Commun., vol. COM-23, pp. 5-15.

- B2 H. Bobeck, P.I. Bonyhard and J.E. Geusic, "Magnetic bubbles - an emerging new memory technology," Proc. IEEE, vol. 63, pp. 1176-1195, Aug. 1975.
- B3 M. Ball, R.M. Van Slyke, I. Gitman, and H. Frank, "Reliability of packet broadcasting in radio networks," IEEE Trans. Ccts. and Syst., vol. CAS-23, pp. 806-814, Dec. 1976.
- C1 W.W. Chu, Ed., Advances in Computer Communications. Dedham, Mass.: Artech House, 1976.
- C2 T. Cosgrove and R.D. Chipp, "Economic considerations for communication systems," IEEE Trans. Commun. Technol., vol. COM-16, pp. 513-525, Aug. 1968.
- C3 K. Chen, "Cable communications policy issues: an overview," IEEE Trans. Systems, Man. and Cybern., vol. SMC-6, pp. 727-734, Nov. 1976.
- C4 D. Chen and J.D. Zook, "An overview of optical data storage technology," Proc. IEEE, vol. 63, pp. 1207-1230, Aug. 1975.
- C5 J.K. Cavers, "Cutset manipulation for communication network reliability estimation," IEEE Trans. Commun., vol. COM-23, pp. 569-575, June 1975.
- C6 W.R. Crowther, F.E. Heart, A.A. McKenzie, J.M. McQuillan, and D.C. Walden, "Issues in packet switching network design," in AFIPS Conf. Proceedings, Nat'l Comput. Conf., pp. 161-175, 1975.
- C7 G.J. Clowes and C.S. Janasuriya, "Traffic considerations in switched data networks," Proc. Third IEEE Symp. on Data Nets. Anal. and Design, St. Petersburg, Fla., Nov. 1973, pp. 18-22.
- C8 V.G. Cerf and R.E. Kahn, "A protocol for packet network inter-communication," IEEE Trans. Commun., vol. COM-22, pp. 637-648, May, 1974.
- D1 A.L. Dudick, E. Fuchs and P.E. Jackson, "Data traffic measures for inquiry-response computer communication systems," in Information Processing 71, C.U. Freiman, Ed., North-Holland: Amsterdam, 1972; also in [C1].
- D2 D.A. Dunn and A.J. Lipinski, "Economic considerations in computer-communication systems," in [A1], ch. 10.
- D3 H. Diriltten and R.W. Donaldson, "Topological design of distributed data communication networks using linear regression clustering," IEEE Trans. Commun., vol. COM-25, Oct. 1975 (in press).

- D4 H. Diriltten and R.W. Donaldson, "Topological design of teleprocessing networks using linear regression clustering," IEEE Trans. Commun., vol. COM-24, pp. 1152-1159, Oct. 1976.
- D5 R.W. Donaldson, "Communications for Text Processing: With Application to Electronic Information Services." Report to the Department of Communications, Ottawa, Canada, Jan. 1977.
- D6 D.W. Davies, C.R. Evans and D.M. Yates, "Human factors in interactive teleprocessing systems," in Conf. Rec., Second Int. Conf. Comput. Commun., Stockholm, Sweden, Aug. 1974, pp. 491-496.
- D7 J. DeMercado, N. Spyrtatos and B.A. Bowers, "A method for calculation of network reliability," IEEE Trans. Rel., vol. R-25, pp. 71-76, June 1976.
- F1 R. Fajman and J. Borgett, "WYLBUR: An interactive text-editing and remote job entry system," Commun. ACM, vol. 16, pp. 314-322, May 1973.
- F2 N. Foy and W. Helgason, "Europe claims the lead in banking," Datamation, vol. 22, pp. 57-59, July 1976.
- F3 E. Fuchs and P.E. Jackson, "Estimates of distributions of random variables for certain computer communication traffic models," Commun. ACM, vol. 13, pp. 752-757, Dec. 1970; also in [C1].
- F4 H. Frank, M. Gerla and W. Chou, Issues in the design of large distributed computer communication networks," in Conf. Rec., Nat'l Telecom. Conf., Atlanta, Ga., 1973, pp. (37A-1)-(37A-9).
- F5 J.D. Foley and V.L. Wallace, "The art of graphics in man-machine conversation," Proc. IEEE, vol. 62, pp. 462-471, Apr. 1974.
- F6 H. Frank, "Survivability analysis of command and control communications networks - Part I," IEEE Trans. Commun., vol. COM-22, pp. 589-595, May 1974.
- F7 H. Frank, "Survivability analysis of command and control communications networks - Part II," IEEE Trans. Commun., vol. COM-22, pp. 596-605, May 1974.
- F8 H. Frank and I.T. Frisch, Communication, Transmission and Transportation Networks. Addison-Wesley, Reading, Mass., 1971.
- F9 M.J. Fisher and T.C. Harris, "A model for evaluating the performance of an integrated circuit-and packet-switched multiplex structure," IEEE Trans. Commun., vol. COM-24, pp. 195-202, Feb. 1976.

- G1 M. Gerla, "New line tariffs and their impact on network design," in Proc. Nat. Comput. Conf., Chicago, Ill., vol. 43, pp. 577-582, 1974.
- H1 G.D. Hodge, "An electronic mail system -- will it happen?" in Conf. Rec., Int. Conf. Comput. Commun., Stockholm, Sweden, Aug. 1974, pp. 351-357.
- H2 D. Hodges, "A review and projection of semiconductor components for digital storage," Proc. IEEE, vol. 63, pp. 1136-1159, Aug. 1975.
- H3 W.C. Hughes, "A semiconductor nonvolatile electron beam accessed mass memory," Proc. IEEE, vol. 63, pp. 1230-1240, Aug. 1975.
- H4 D.L. Hebditch, "Terminal systems for real people," in Conf. Rec., Second Int. Conf. Comput. Commun., Stockholm, Sweden, Aug. 1974, pp. 497-504.
- H5 E. Hänsler, G.K. McAuliffe and R.S. Wilkov, "Exact calculation of computer network reliability," AFIPS Proceedings, vol. 41, 1972; also in [C1].
- H6 E. Hänsler, and G.V. McAuliffe, "A fast recursive algorithm to calculate the reliability of a communication network," IEEE Trans. Commun., vol. COM-20, pp. 637-640, June 1972.
- I1 K. Itoh and T. Kato, "An analysis of traffic handling capacity of packet switched and circuit switched networks," in Proc. Third IEEE Symp. Data Nets. Anal. and Design, St. Petersburg, Fla., Nov. 1973, pp. 29-37.
- J1 C.J. Jenny and K. Kummerle, "Distributed processing within an integrated circuit/packet-switching node," IEEE Trans. Commun., vol. COM-24, pp. 1089-1100, Oct. 1976.
- J2 F.J. Jensen, "Centralization or decentralization in banking," Datamation, vol. 22, p. 60, July 1976.
- K1 C.R. Kraus, "Meeting this public's communication needs," in Conf. Rec., IEEE Int. Conf. Commun., Seattle, Wash., 1973, pp. (6-1)-(6-6).
- K2 L. Kleinrock, Communication Nets: Stochastic Message Flow and Delay. New York, N.Y.: McGraw-Hill, 1964.
- K3 L. Kleinrock, and W.E. Naylor, "On measured behaviour of the ARPA network," in Proc. Nat. Comput. Conf., vol. 43, Montvale, N.J.: AFIPS Press, 1974, pp. 767-780.
- K4 L. Kleinrock, Queueing Systems, Vol. 2: Computer Applications. New York, N.Y.: Wiley, 1976.

- K5 T.C.S. Kennedy, "The design of interactive procedures for man-machine communication," *Int. J. Syst. Science*, vol. 6, pp. 309-334, May 1974.
- K6 A. Kershenbaum and R.M. Van Slyke, "Recursive analysis of network reliability," *Networks*, vol. 3, pp. 81-94, 1973.
- K7 H. Kobashi and A.G. Konheim, "Queueing models for computer communications system analysis," *IEEE Trans. Commun.*, vol. COM-25, pp. 2-29, Jan. 1977.
- K8 A.G. Konheim, "Chaining in a loop system," *IEEE Trans. Commun.*, vol. COM-24, pp. 203-209, Feb. 1976.
- L1 C.M. Laucht, "Electronic mail for the Canadian environment," in *Conf. Rec., IEEE Int. Conf. Commun.*, Philadelphia, Penn., 1976, pp. (15-3)-(15-4).
- L2 P.M. Lin, B.J. Leon and T.C. Huang, "A new algorithm for symbolic system reliability analysis," *IEEE Trans. Rel.*, vol. R-25, pp. 2-15, April 1976.
- M1 J. Martin and A. Norman, The Computerized Society. Englewood Cliffs, N.J.: Prentice-Hall, 1973.
- M2 W.J. Miller, "Technology assessment for electronic message handling," in *Conf. Rec., IEEE Int. Conf. Commun.*, Philadelphia, Penn., 1976, pp. (15-10)-(15-12).
- M3 P.U. McEnroe, H.T. Huth, E.A. Moon and W.W. Morris III, "Overview of the supermarket system and the retail store system," *IBM Systems Journal*, vol. 14, pp. 3-15, Jan. 1975.
- M4 J. Martin, Systems Analysis for Data Transmission. Englewood Cliffs, N.J.: Prentice-Hall, 1972.
- M5 J.C. Mallinson, "Tutorial review of magnetic recording," *Proc. IEEE*, vol. 64, pp. 126-208, Feb. 1976.
- M6 J. Martin, Design of Man-Computer Dialogues. Englewood Cliffs, N.J.: Prentice-Hall, 1973.
- M7 H. Miyahara, T. Masagawa and Y. Teshigawara, "A comparative analysis of switching methods in computer communications," in *Conf. Rec., IEEE Inter. Conf. Commun.*, San Francisco, Calif., June 1975, pp. (6-6)-(6-10).
- N1 D.L. Neuhoff, "The Viterbi algorithm as an aid in text recognition," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 222-226, March, 1975.
- N2 J.M. Nilles, F.R. Carlson, P. Gray and G. Hanneman, "Telecommuting - an alternative to urban transportation congestion," *IEEE Trans. Systems, Man. and Cybern.*, vol. SMC-6, pp. 77-85, Feb. 1976.

- 01 S.S. Oren, "A mathematical theory of man-machine document assembly," IEEE Trans. on Systems, Man, Cybern., vol. SMC-5, pp. 520-527, Sept. 1975.
- 02 S.S. Oren, "A mathematical theory of man-machine text editing," IEEE Trans. on Systems, Man, Cybern., vol. SMC-4, pp. 258-267, May 1974.
- 03 S. Osaki and T. Nakagawa, "Bibliography for reliability and availability of stochastic systems," IEEE Trans. Rel., vol. R-25, pp. 284-287, Oct. 1976.
- P1 G.E. Passant, "Operational impact of real time computing in trustee savings banks," in Conf. Rec., Int. Conf. Comput. Commun., Stockholm, Sweden, Aug. 1974, pp. 83-85.
- P2 R.R. Panko, R.W. Hough, and R. Pye, "Telecommunications for office decentralization: apparent needs and investment requirements," in Conf. Rec., IEEE Int. Conf. Commun., Philadelphia, Penn., June 1976, pp. (26-8)-(26-13).
- P3 R. Pye and P.I. Weintraub, "Social and organizational implications of decentralization," in Conf. Rec., IEEE Int. Conf. Commun., Philadelphia, Penn., June 1976, pp. (26-14)-(26-19).
- P4 A. Papoulis, Probability, Random Variables and Stochastic Processes. New York, N.Y.: McGraw-Hill, 1965.
- R1 M.A. Robbins, "Error objective in the electronic mail system," in Conf. Rec., IEEE Int. Conf. Commun., Philadelphia, Penn., 1976, pp. (15-7)-(15-9).
- R2 L. Roberts, "Extensions of packet communication technology to a hand held personal terminal," in AFIPS Conf. Proceedings, SJCC, vol. 40, Montvale, N.J.: AFIPS Press, 1972, pp. 295-298; also in [C1].
- R3 R.D. Rosner, "Cost considerations for a large data network," in Conf. Rec., Second Inter. Conf. Comput. Commun., Stockholm, Sweden, Aug. 1974, pp. 289-298.
- R4 R.D. Rosner, R.H. Bittel and D.E. Brown, "A high throughput packet-switched network technique without message reassembly," IEEE Trans. Commun., vol. COM-23, pp. 819-828, Aug. 1975.
- R5 L.G. Roberts, "Data by the packet," IEEE Spectrum, vol. 11, pp. 46-51, Feb. 1974.
- R6 W.B. Rouse, "Design of man-computer interfaces for on-line interactive systems," Proc. IEEE, vol. 63, pp. 817-857, June 1975.
- R7 I. Rubin, "Message delays in packet-switching communication systems," IEEE Trans. Commun., vol. COM-23, pp. 186-192, Feb. 1975.

- S1 B. Skoldborg, "Real time banking system as an application," in Conf. Rec., Int. Conf. Comput. Commun., Stockholm, Sweden, Aug. 1974, pp. 71-82.
- S2 M. Schwartz, Computer Communication Network Design and Analysis. Englewood Cliffs, N.J.: Prentice-Hall, 1977.
- S3 A.P. Sage, A Methodology for Large Scale Systems Analysis. New York, N.Y.: McGraw-Hill, 1977.
- S4 C.A. Sunshine, "Interprocess Communication Protocols for Computer Networks," Stanford Elect. Lab. Tech. Rept., Stanford Univ., Stanford, Calif., Dec. 1975.
- T1 D.J. Theis, "Communication processors," Datamation, pp. 31-44, Aug. 1972.
- T2 M. Tainita, "A new deterministic network reliability measure," Networks, vol. 6, pp. 191-205, July 1976.
- V1 R. Van Slyke and H. Frank, "Network reliability analysis: Part I," Networks, vol. 1, pp. 279-290, 1972; also in [C1].
- W1 R.L. Williams, "Mailgram -- an electronic mail service," in Conf. Rec., IEEE Int. Commun., Philadelphia, Penn., 1976, pp. (15-5)-(15-6).
- W2 R.S. Wilkov, "Analysis and design of reliable computer networks," IEEE Trans. Commun., vol. COM-20, pp. 666-678, June 1972; also in [C1].
- Z1 G.K. Zipf, Human Behaviour and the Principle of Least Effort. Reading, Mass.: Addison-Wesley, 1949.

III MESSAGE QUEUING AND CHANNEL SHARING

III-1 Introduction and Overview

The primary purpose of this chapter is to present results which enable delays experienced by single messages or packets in an EIS network to be calculated.

Because messages arrive at nodes at irregular and randomly spaced time intervals, queues may form. Delays due to queuing then ensue, even though the average message rate is less than the channel capacity. Section 3-3 presents equations whereby queuing delays and delay variances can be obtained for Poisson message arrivals with arbitrary message length distributions.

Different message streams may share a data channel, in which case multiplexing is required. Section 3-4 considers non-contention multiplexing, including frequency division and time division multiple access. Spread spectrum and pulse address multiple access are also considered, although these involve some degree of contention. Non-contention schemes are seen to be unsuited to EIS applications which generate highly bursty traffic.

Contention multiplexing implies that various users or message streams attempt to utilize a channel with little or no regard for the behaviour of other users. As a result, users may "collide", and each must retransmit his message, hopefully in a manner which prevents another collision. Section 3-5 summarizes delay and throughput results for pure and slotted ALOAH contention multiplexing.

Various strategies have been proposed to reduce delay and throughput degradations due to contention "collisions". Some of

these are considered in Section 3-6. Further study is warranted in this area.

Section 3-7 deals with the interaction of queuing delays and contention collisions, a subject which has not received much attention. Section 3-8 discussing polling, a scheme whereby each user sharing a channel does not transmit until so instructed. Polling avoids collisions but often at the expense of reduced throughput and increased delay.

No one accessing scheme is best in all situations. Contention schemes are best in light traffic, particularly when link propagation delays are large in comparison with message or packet length. Heavy traffic favours modified contention multiplexing or polling. The scheme which is best will depend on the specific channel and traffic parameters.

Modems are important components of any data communication system, and are considered briefly in Section 3-2. Although the discussion is necessarily brief, the important problems including signal and receiver design, synchronization and implementation are considered.

III-2 Modems

"Modem" is an acronym for modulator-demodulator. A modulator converts symbols or more generally, symbol sequences into signals for transmission over a physical communication channel. A demodulator converts received signals into received symbol sequences.

A tradeoff exists between the speed and accuracy with which symbols can be transmitted $[W1, L1, L2, R1]$. When channel band-

width W is narrow in the sense that the symbol rate is comparable to, or larger than W^{-1} , multi-amplitude signalling may be used to reduce the channel band rate. The number of usable amplitude levels depends on the desired transmission error probability and on the signal-to-noise ratio at the receiver input. When the symbol rate is much less than W^{-1} , redundant channel coding can be used to make full use of the available bandwidth, thereby reducing data transmission errors $[L1, L2, L3, W1]$.

Fig. 3-1 shows an often used linear time-invariant model for a physical communication channel. The filter $H(f)$ may actually be present, or may be implicit in a bandwidth constraint on the modulator output signal $s(t)$. Noise $n(t)$ is often modelled as a wide-sense stationary Gaussian random process $[W1, L1, L2]$.

Consider a sequence of data symbols $\dots a_N \dots a_0 \dots a_N \dots$, and a pulse shape $p(t)$. Define

$$g(t) = \sum_{k=-\infty}^{\infty} a_k p(t-kT) \quad (3-1)$$

where T is the basic symbol period. The transmitted signal $s(t)$ in Fig. 3-1 may be either amplitude or angle modulated by $g(t)$. The data symbols a_k may be identical to the original message symbols, or may be subdivisions of these as when each message symbol is represented by a unique bit sequence, or may be combinations of the original sequences.

The modem design problem is easily stated: specify pulse shape $p(t)$ in (3-1) and the receiver to operate on the received signal $r(t)$ to minimize the probability of a symbol error at a specified data rate. Alternatively the data rate may be maximized

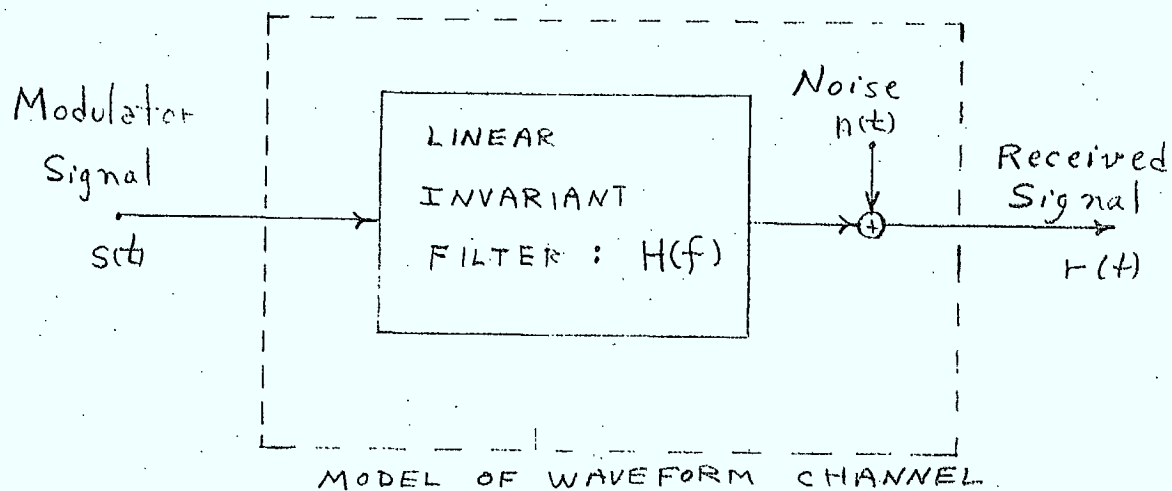


Fig. 3-1 Model of a physical communication channel.

for a specified error probability. Either or both of the peak or average power of $s(t)$ is normally constrained.

During the past decade rapid and sustained progress has been achieved on AM modem design [B1, T1, S1, F1, F2, F3, L1, M1, C1]. In 1967 Berger and Tufts [B1, T1] determined the best pulse shape and linear receiver, the latter a lumped filter followed by a tapped delay line which, in practice uses a finite number of taps. Salz [S1] used (non-linear) decision feedback to improve the performance of the linear equalizer. Care is required to ensure receiver stability [C2]. Recently a tapped delay line followed by a Viterbi decoder [F3, F4] has been used to further improve performance to the point where multi-amplitude quadrature PAM has been used to obtain bit rates in excess of 12 Kbs and bit error rates of 10^{-3} or better on standard telephone channels with approximately 2800 Hz of useable bandwidth [F2, F1, M1].

The number of useable PAM levels depends on the signal-to-noise ratio at the receiver; a tradeoff exists between the number levels and the baud rate T in (3-1). For channels satisfying a Nyquist distortion criterion [L1], the symbol error probability P_e can be expressed in terms of the number of symbol levels L (the number of discrete values assumed by a_k in (3-1)) and the ratio of the received signal-to-noise power S/N as follows:

$$P_e = 2\left(1 - \frac{1}{L}\right) Q\left(\sqrt{\frac{3}{L^2 - 1}} \sqrt{\frac{S}{N}}\right)^{1/2} \quad (3-2)$$

where

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-y^2/2} dy \quad (3-3)$$

In contrast with the AM case, little progress has been achieved in optimizing either the pulse shape $p(t)$ or the receiver for angle modulation. The difficulties arise from the problems inherent in determining spectra for angle modulated signals [L1].

In line-switched networks, the response $H(f)$ in Fig. 3-1 is not known prior to the establishment of a circuit, with the result that adaptive receivers are often used to achieve high data rates [L1, C1, D1]. In linear equalizers the delay line taps are adjusted adaptively using a steepest descent or other suitable algorithms [L1, C1, D1]. The Viterbi algorithm is inherently adaptive [F4].

One of the most difficult problems in data communications is synchronization, which is required at the symbol and word level [S2] and ultimately at the process level. Carrier synchronization can be avoided by use of incoherent detection techniques [L1, L2, W1, S2]. Symbol synchronization involves sampling the receiver output at a time which will minimize the error probability [M2, L1, L2, S2, U1]. Word synchronizations are considered in Chapter 5 in conjunction with data link control.

When the data rate is sufficiently low that intersymbol-interference is negligible, symbol synchronization is rather easily achieved by correlating the received data stream with time-shifted replicas of the modulator pulse [L2, S2, S3]. The time-shifting and correlation can be implemented either serially or in parallel [L2]. When intersymbol-interference is present, symbol synchronization is much more difficult, and normally involves use of zero-

crossings or maximum eye openings [L1, M2, D1, U1, Q1].

Adaptive equalization and symbol synchronization are of particular interest to designers of interactive data communication systems because of time delays inherent in these operations. When half-duplex data links are used, the above time delays can occupy from 100 to 200 ms [D1], which is in addition to the 150-200 ms needed to reverse echo suppressors [D1]. The future may see more effort to design signals and receivers to reduce symbol synchronization times, possibly at the expense of slightly lower data rates. The time required to establish link data transmission motivates use of dedicated full-duplex channels.

Considerable effort is being expended to reduce the cost of modem implementation [H1, V1, D1]. Efforts are also being expended in modelling fibre optic channels to facilitate design of suitable modems [G1, H2, M3, P1, R2, T2].

III-3 Message Queuing at Network Nodes

In this section queuing theory results of immediate concern to our study are summarized. Queuing theory and its applications are discussed at length in various references [K1, K2, K3, R3, S4, K4, S5, S6].

Fig. 3-2 shows a network node fed by messages from input channels 1 to n . The average message arrival rate on input channel i and departure rate on output channel j are ζ_i and λ_j , respectively. The message lengths are y_i and x_j , as shown.

The node in Fig. 3-2 may represent a buffered data terminal operating in transmit mode, in which case $n=1$, a receiving terminal in which case $m=1$ or an internal network node with $n>1$ and $m>1$.

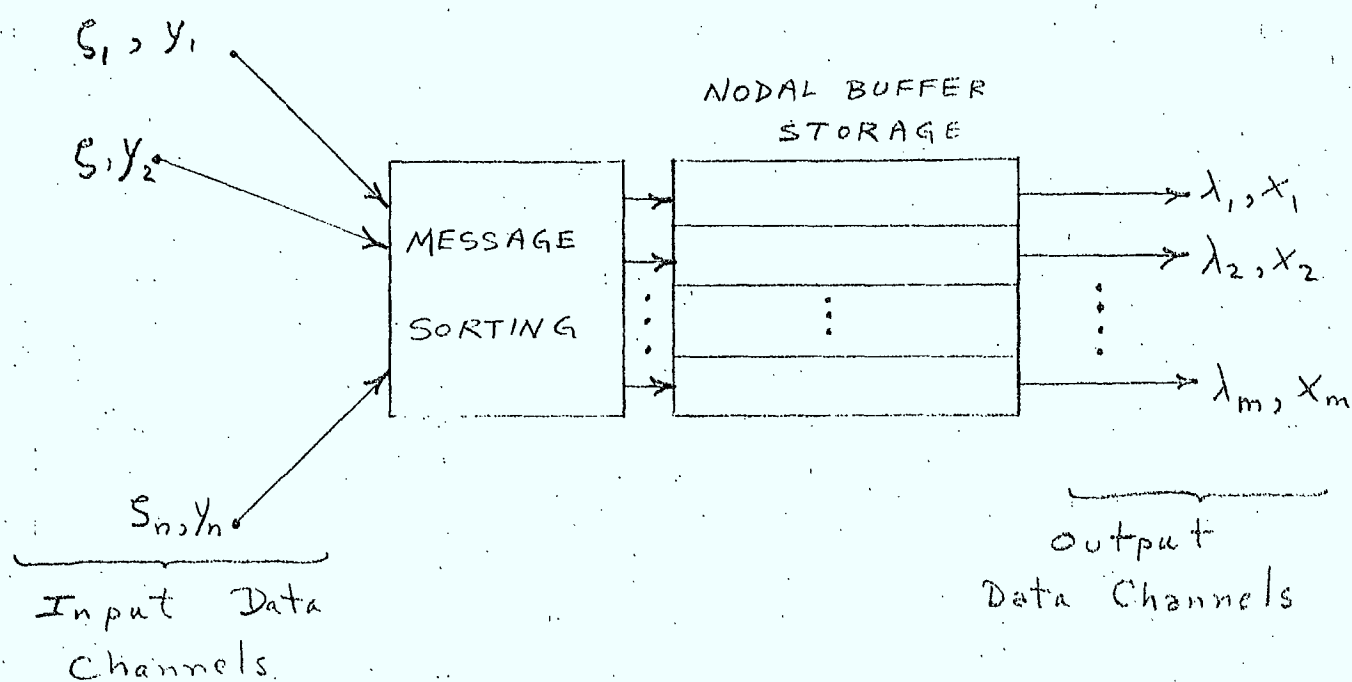


Fig. 3-2 Queuing of messages at a network node.

Messages arriving on any incoming channel i will normally be destined for different output channels. Infinite nodal buffer storage is assumed. Effects of finite buffers are considered in Sections 4-10 and 5-7.

The average waiting time for the queued messages and the waiting time variance depend on message arrival and length distributions and the queue service discipline. If the incoming message arrival rates are Poisson, the messages destined for each output channel also have Poisson arrivals with rate $\lambda_i = \lambda$ where

$$\lambda = \sum_{j=1}^m h_j \zeta_j \quad (3-4)$$

where h_j is the fraction of messages on input channel j destined for output channel i [$K1, K2, P2$].

The average time that a message waits in the queue for service is [$K2, M4, S5$]

$$W = \lambda \bar{x}^2 / 2(1-\rho) \quad (3-5)$$

$$= [\rho \bar{x} / (1-\rho)] [\bar{x}^2 / 2\bar{x}^2] \quad (3-6)$$

where \bar{x} denotes the mean value of message length $x_i = x$, \bar{x}^2 = the mean squared length and

$$\rho = \lambda \bar{x} \quad (3-7)$$

The variance σ_w^2 of the waiting time W is [$K2$]

$$\sigma_w^2 = w^2 + \lambda \bar{x}^3 / 3(1-\rho) \quad (3-8)$$

$$= w^2 + [\rho \bar{x}^2 / (1-\rho)] [\bar{x}^3 / 3\bar{x}^3] \quad (3-9)$$

The average message delay T includes the average time required to remove the message from the queue in Fig. 3-2. Thus

$$\begin{aligned} T &= \bar{x} + W \\ &= \bar{x} [1 + (\rho / (1-\rho))] [(x^2 / 2\bar{x}^2)] \end{aligned} \quad (3-10)$$

and

$$\sigma_T^2 = \bar{x}^2 + \sigma_w^2 \quad (3-11)$$

where σ_T^2 is the variance of T .

As expected, T and W are proportional to \bar{x} , while σ_w^2 and σ_T^2 are proportional to \bar{x}^2 . It is seen that T , W , σ_T and σ_w all increase rapidly as $\rho \rightarrow 1$. Thus, a large utilization factor ρ not only increases the average waiting and delay times, but also increases the variance of these times.

Equations (3-6) and (3-10) apply to any priority queuing discipline which selects messages independently of the message lengths [K2]. Included in this class is the familiar first-come-first-served (FCFS) discipline. Equations (3-8) and (3-10) apply only to the FCFS discipline, however. Assignment of high priorities to short messages reduces W [K2, M5]. The reduction is particularly when ρ is close to unity [M5].

Knowledge of either W or T and the corresponding variance enables determination of the probability that the instantaneous delay t or waiting time w will exceed a specific value, provided that t

or w can be approximated by a known probability distribution $f_t(u)$ or $f_w(u)$. If $P(t < A)$ and $P(w < B)$ denote, respectively, the probability that delay $t < A$ and waiting time $w < B$ then

$$P(t < A) = \int_0^A f_t(u) du \quad (3-12)$$

$$P(w < B) = \int_0^B f_w(u) du \quad (3-13)$$

Martin [M4] argues that $f_t(u)$ and $f_w(u)$ are well approximated by Gamma distributions with parameter $R = (T/\sigma_T)^2$, and the traffic studies cited in Chapter 2 tend to support this assumption.

Knowledge of $f_t(u)$ is of interest in estimating the amount of buffer capacity required at each node. (See Section 5-7).

Distributions $f_x(u)$ of the message length $x_i = x$ in Fig. 3-2 can be expressed in terms of the component message length distributions $f_{\zeta_j}(u)$ as follows:

$$f_x(u) = \sum_{j=1}^n \frac{\zeta_j}{\lambda} f_{\zeta_j}(u) \quad (3-14)$$

It follows that

$$\overline{x^k} = \sum_{i=1}^n \frac{\zeta_j}{\lambda} \left[\overline{\zeta_j^k} \right] f_{\zeta_j}(a) \quad (3-15)$$

From (3-14) and (3-15) it follows that if all input channels have identical message length distributions, $f_x(u)$ and $\overline{x^k}$ are identical to $f_{\zeta_i}(u)$ and $\overline{\zeta_j^k}$, respectively, for $j=1,2,\dots,n$.

Both T in (3-10) and W in (3-6) depend on the ratio $\phi = \overline{x^2} / \bar{x}^2$ which is shown in Fig. 3-3 for various message length


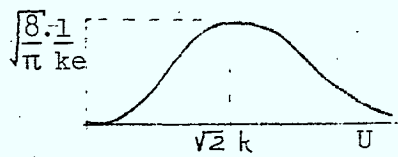
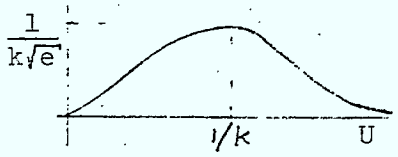
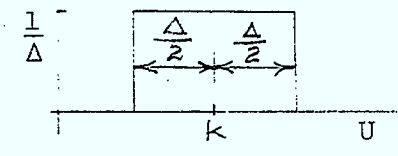
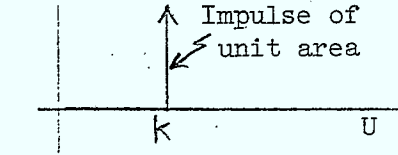
Form of $f_x(U)$	Mathematical Form for $U \geq 0$	Sketch of $f_x(U)$	\bar{x}	\bar{x}^2	$\phi = \bar{x}^2 / \bar{x}^2$
Exponential	$f_x(U) = k^{-1} e^{-U/k}$		k	$2k^2$	2.0
Maxwell	$f_x(U) = \sqrt{\frac{2}{\pi}} \frac{U^2}{k^3} e^{-U^2/2k^2}$		$\sqrt{\frac{8}{\pi}} k$ $\approx 1.60 k$	$3k^2$	1.18
Rayleigh	$f_x(U) = \frac{U}{k^2} e^{-U^2/2k^2}$		$\sqrt{\frac{\pi}{2}} k$ $\approx 1.25 k$	$2k^2$	1.27
Uniform	$f_x(U) = \begin{cases} \frac{1}{\Delta} & ; k - \frac{\Delta}{2} < U < k + \frac{\Delta}{2} \\ 0 & ; \text{all other } U \end{cases}$		k	$k^2 + \frac{\Delta^2}{12}$	$1 + \left(\frac{\Delta}{k}\right)^2 \frac{1}{12}$
Constant	$f_x(U) = \delta(U-k)$		k	k^2	1.0

Fig. 3-3: \bar{x} , \bar{x}^2 and ϕ for various message length distributions.

distributions. Ratio Φ is seen to vary from 1.0 for a constant distribution to 2.0 for the exponential distribution. The exponential distribution is often used to calculate delays in computer-communication networks in which case waiting time W_e and delay T_e are as follows (See also Fig. 3-4):

$$W_e = \rho \bar{x} / (1-\rho) \quad (3-16)$$

$$T_e = \bar{x} / (1-\rho) \quad (3-17)$$

Since $\overline{x^3} = 6 \bar{x}^3$ for exponential message lengths

$$\sigma_{W_e}^2 = W_e^2 + 2\rho \bar{x}^2 / (1-\rho) \quad (3-18)$$

$$\sigma_{T_e}^2 = \overline{x_e^2} + \sigma_{W_e}^2 \quad (3-19)$$

Recall from Section 2-2 that the Gamma distribution, by appropriate choice of its two parameters b and c could represent both the Gamma and exponential distribution, and that for all choices of b and c , $1 \leq \Phi \leq 2$. Thus, it is not unreasonable to expect that for unimodal distributions W and T are largest for the exponential distribution and smallest for constant message length. Since exponential message lengths are often assumed in calculating T , the actual value of T will often be smaller.

In those situations where the node in Fig. 3-1 is an internal network node, the delay seen by messages destined for another node includes in addition to T in (3-10) a propagation delay P and nodal

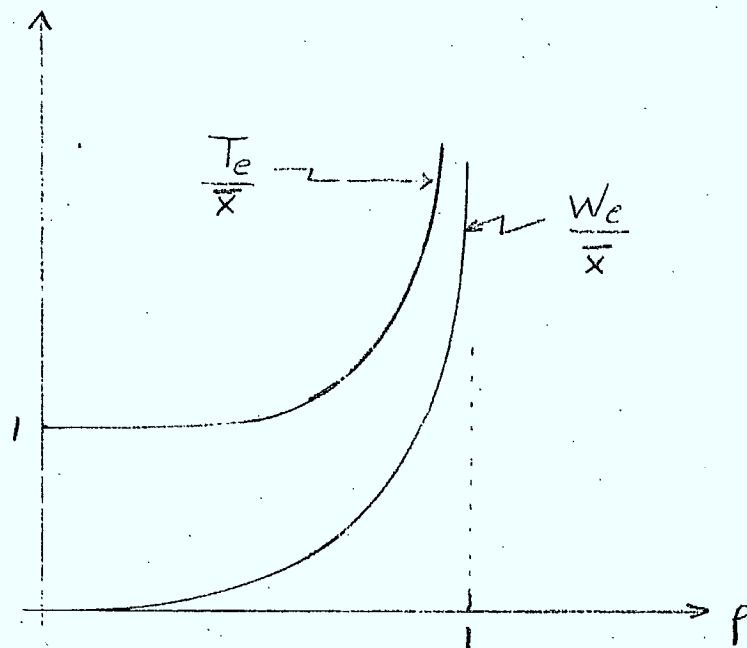


Fig. 3-4 Waiting time W_e/\bar{x} and delay T_e/\bar{x} for exponential message lengths.

processing delay K . If significant overhead information is transmitted along with the actual data over an output channel of capacity C , and if μ^{-1} and $(\mu^1)^{-1}$ denote, respectively, the average length in bits of data packets and all packets (including data and control packets) then

$$T = W + \frac{1}{\mu C} + P + K \quad (3-20)$$

and

$$W = \left[\frac{\lambda / \mu^1 C}{\mu^1 C - \lambda} \right] \frac{\Phi}{2} \quad (3-21)$$

If exponentially distributed message lengths are assumed for all packets, then (3-20) reduces to Kleinrock's [K2] formula, since $\Phi = 2$ in this case.

The above discussions assume continuous message lengths. In those situations where message lengths are short ($\simeq 50$ bits) discrete length distributions are used, and the above results are easily modified [S5] as indicated in Section 2-2.

III-4 Non-Contention Multiplexing

Time division multiple access (TDMA) and frequency division multiple access (FDMA) are non-contention multiplexing schemes, since dedicated time slots or frequency bands, respectively, are available for communication. We also include pulse address multiple access (PAMA) and spread spectrum multiple access (SSMA) under non-contention multiplexing, even though a user's transmissions may be subject to severe interference when many other users are active [S7, S8].

Addressing is inherent in all four schemes. In FDMA and

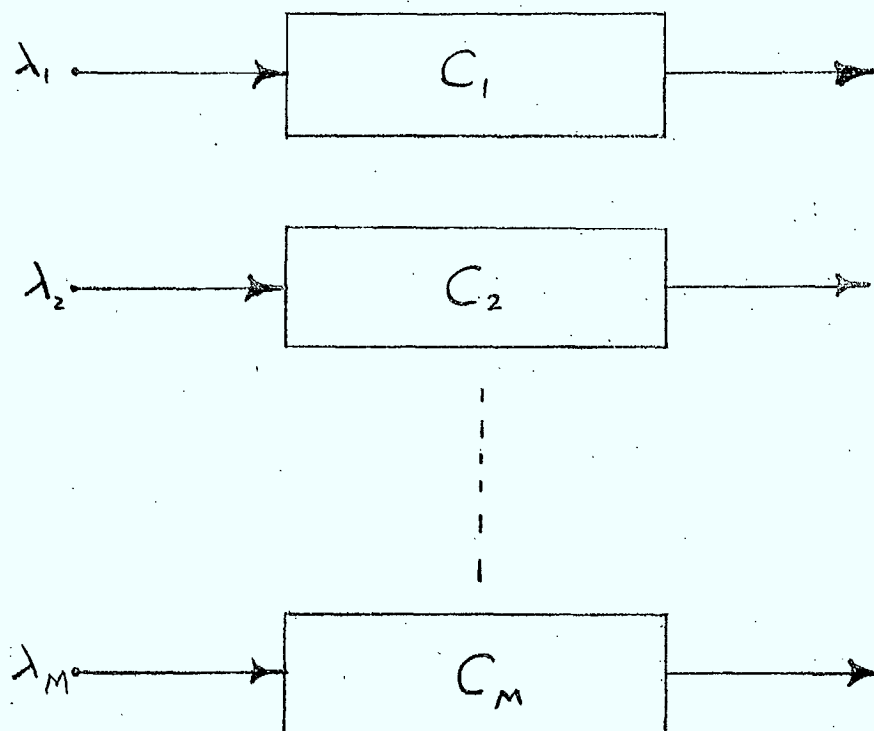
TDMA the demodulator monitors the appropriate frequency channel or time slot, respectively. In SSMA, demodulation is via a particular pseudo-random waveform, and in PAMA via a sequence of pulses with a distinct time-frequency pattern [S7, S8]. SSMA and PAMA are particularly suited for transmission security and privacy; jamming of a particular frequency band will not obliterate transmission and guessing the correct pseudo-random waveform or pulse sequence is very unlikely. The sender is also identified by the frequency (in FDMA), time slot (in TDMA), pseudo-noise waveform (in SSMA) and pulse pattern (in PAMA).

FDMA offers implementation simplicity (network timing is not required) and compatibility with much existing equipment. When non-linearities are involved, as in satellite repeaters operating in saturation, intermodulation products reduce the useable repeater output power in which case some co-ordination may be required among up-link user power.

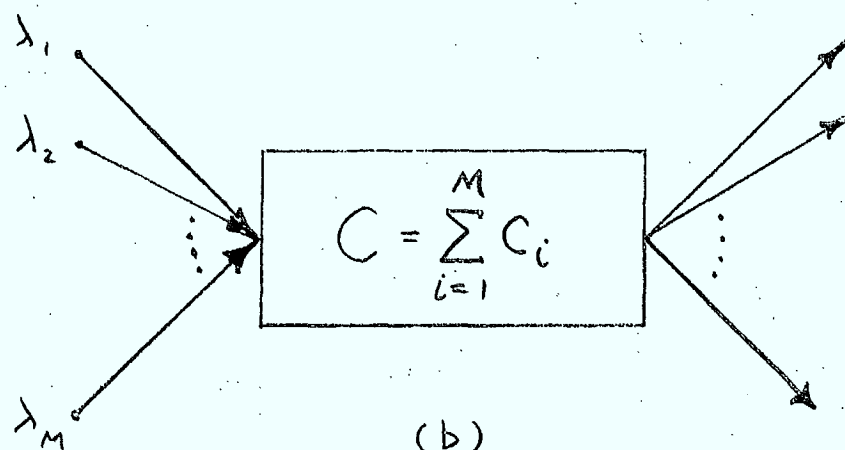
TDMA requires network timing, however, mutual interference among various users' signals is obviated, as is the need for power co-ordination.

Neither FDMA nor TDMA are particularly suited to bursty, low-duty-cycle EIS data traffic. Fig. 3-5 (a) shows M separate FDMA or TDMA channels of capacity C_i ; Fig. 3-5 (b) shows a single shared channel of capacity C . If all message arrivals are Poisson, if $\lambda_i = \lambda$ and $C_i = C$ for all $i=1,2,\dots,M$, and if all messages have the same length distribution then (3-10) yields delays T_a and T_b for Figs. 3-5 (a) and (b) as follows:

$$T_a = (M/\mu C)(1 + [\rho/(1-\rho)] \Phi/2) \quad (3-22)$$



(a)



(b)

Fig. 3-5 Illustrating FDMA and TDMA (a) M separate channels of capacity C/M . (b) Single shared channel of capacity C .

$$T_b = T_a/M \quad (3-23)$$

$$\rho = \lambda/\mu C \quad (3-24)$$

From (3-22) to (3-24) it follows that if all messages queue for one shared high-capacity channel, rather than for several dedicated low-capacity channels of equal total capacity, then average message delay is reduced. This statement applies for arbitrary values of λ_i and C_i , provided delay T is averaged over all M message sources. Analysis shows that TDMA, provides a slightly smaller delay T than FDMA, at the expense of additional circuitry for timing control [T3].

SSMA involves multiplication of a narrow-band data waveform by a wide-band pseudo-random binary waveform, followed by translation of the composite signal via amplitude or angle modulation to the desired frequency band [A1, S7, S8]. Demodulation involves conversion of the received signal to the baseband followed by multiplication using the same pseudo-random waveform employed at the transmitter. The resulting signal is then matched-filtered prior to detection of each data bit. If the data bits are rectangular pulses then the matched filter is a simple integrator.

SSMA noise is caused mainly by other users whose wideband signals appear as noise to the user in question. Thus, if there are $M-1$ other users, if the total received power is P , if N is the thermal noise power spectral density at the receiver, if W and B denote, respectively, the system and message bandwidth then

for equal user transmit powers and no intersymbol interference, detectability d^2 is as follows [S7, S8]:

$$d^2 = \frac{K}{[(M-1)/M]} \frac{2(P/MB)}{(P/W)+N} \quad (3-25)$$

where constant $K = 1$ for a linear channel and $K = \Pi/4$ for a hard-limiter channel. For large M , $(M-1)/M \approx 1$ with the result that

$$d^2 \approx \frac{2K}{M} \frac{(P/N)(W/B)}{W + P/N} \quad (3-26)$$

For coherent detection, the bit-error probability p is as follows:

$$p = \frac{1}{\sqrt{2\pi}} \int_d^{\infty} \exp(-y^2) dy \quad (3-27)$$

For incoherent detection of the carrier signal

$$p = (1/2) \exp(-d/2) \quad (3-28)$$

Increasing W/B increases d and reduces p .

Results which include intersymbol interference are also available [H3].

SSMA does not require centralized timing control. Some user power co-ordination is required to prevent low power users from encountering unacceptably low detectabilities. This statement applies particularly to hard-limiter channels, since strong signals tend to capture the limiter and weak signals suffer a factor of four suppression beyond that for a linear repeater [A1, S7, S8].

SSMA channels are characterized by a time-variation in the number of users M and therefore in error probability p which varies as $1/M$. Proposed remedies include variation of the message bit duration $1/B$ [C3, H4, C4]. Each user either monitors his own transmissions or those of others and thereby observes the detection reliability, or is so advised via a return channel. The user then adjusts his message rate accordingly. When satellite channels are involved the 0.26 sec. round-trip delay requires a 0.13 sec. prediction of the channel occupancy. If the occupancy variation rate greatly exceeds $0.13^{-1} \approx 7.7/\text{sec.}$, then prediction accuracy is poor. ARPANET [K2] packet duration is 22.5 msec., with the result that channel occupancy prediction is required 12 packet slots in advance; an impossibility. Typically, round trip time for radio networks is approximately 0.05 msec., or 0.5 percent of a 1000-bit packet at rate 100 Kbs. Here occupancy prediction is viable.

Finite buffer effects at transmitter and receiver must also be considered in selecting SSMA transmission rates. In particular, rates which cause buffer overflow or underflow must be avoided [C4]. Finite buffer size may seriously limit performance [C4].

By increasing their transmit power, a few SSMA users could ensure reliable transmission at all times, at the expense of lower power users. Some users would probably be prepared to pay for this high power priority.

Attractive SSMA features include assured channel usage without regard for other users, albeit at possibly low data or

high error rates for which clear tradeoff exists. Jamming and unauthorized decoding of transmissions is difficult. Implementation complexity is moderate.

Disadvantages of SSMA include synchronization delays in high noise environments and inefficient channel usage if $P/NW \gg 1$ [S8]. The latter objection is overcome by using multi-amplitude signalling [L1] as explained in Section 3-2 or by designing signals and receivers to combat intersymbol interference [H3]. Synchronization time increases would occur at low data rates which could tolerate such increases, or in high error rate environments which would require retransmissions even with rapid synchronization. The utility of SSMA for EIS multiplexing seems an open question.

In PAMA each symbol is represented by a sequence of N pulses in one of B different frequency bands. Thus, if the pulses are binary, there are 2^{NB} different combinations. A user may be assigned one pulse sequence for a mark and another for space. PAMA systems offer flexibility and relative implementation simplicity. User power co-ordination and timing are not required. However, timing can increase the number of users which can be active at any one time; if two or more pulses in a given frequency slot overlap in time, the overlap may destroy all pulses involved in the overlap, and possibly each user's entire pulse sequence unless error-correction is used. Without user timing co-ordination overlaps will occur whenever pulses are separated by less than $2D$ sec. where D is the pulse width. If all pulses are bit-synchronized then pulses will either overlap completely, or not at all [K2, K5], with the result that the required pulse separation interval is reduced to D [K2, A3].

As PAMA users increase in number beyond a certain limit, throughput decreases as more messages require retransmission. If too many users attempt access retransmissions increase and the throughput drops to zero, as explained next in considering ALOAH accessing [K2, K5, A3] which is a specialized PAMA scheme.

III-5 Contention Multiplexing

In contention multiplexing users access a channel and retransmit following "collisions" with other users.

Let $q=1-p$ be the successful transmission probability and G the Poisson rate of generation of packets per packet duration D . Since collisions will again occur with certainty if all retransmissions are scheduled exactly τ sec. later, actual retransmission time is varied about the nominal time interval τ [L4, L5].

Assume that any user in an infinite population has at most one packet awaiting retransmission. Waiting time Q is calculated as follows:

$$\begin{aligned} Q &= \tau qp + 2\tau qp^2 + \dots + i\tau qp^i + \dots \\ &= \tau qp \frac{d}{dp} \sum_{i=1}^{\infty} p^i \\ &= \tau p/q \end{aligned}$$

Thus,

$$Q/\tau = p/(1-p) = (1-q)/q \quad (3-29)$$

Similar analysis yields

$$\sigma_Q^2 = \tau^2 \left(\frac{p}{1-p} \right) \left(\frac{p}{p^2 - p + 2} \right) - W^2 \quad (3-30)$$

As expected $Q \rightarrow 0$ as $p \rightarrow 0$.

For a slotted ALOAH channel with messages (or packets) of length D and an infinite user population $[A_1, K_2]$, $q = e^{-G}$ where G includes both original and retransmitted packets. For pure ALOAH, $q = e^{-2G}$. Packet throughput $S = f(G)$ where $f(G)$ equals Ge^{-G} for slotted ALOAH and Ge^{-2G} for pure ALOAH. (In slotted ALOAH, users seek access only at fixed time intervals of width D ; in pure ALOAH transmission occurs at any time $[A_3, K_2]$.) For slotted ALOAH, S is maximized for $G=1$ in which case $S=1/e$. For stable operation $G \leq 1$, in which case $S \leq 1$ and $S=f(G)$ increases monotonically with G . Thus

$$Q/\tau = \exp(f^{-1}(S)) - 1 \quad (3-31)$$

For pure ALOAH, stability requires $G \leq 0.5$ which makes $S \leq 1/2e$.

As expected Q increases with G , slowly at first and then more rapidly as $G \rightarrow 1$. If $G > 1$ for a time, then the collision probability increases to the point where throughput falls. Normally G varies with time; and the way in which G varies determines whether retransmission backlogs clear, or increase to drive the throughput to zero. ALOAH channel stability is considered in Chapter 5 in conjunction with flow control.

Message delay $T = d + Q$ where d is the data link propagation time. For a satellite channel $d \approx 0.26$ sec.

The above analysis excludes delays resulting from randomizing the retransmission interval. Consider slotted ALOAH with retransmission during one of K randomly selected time slots of width D following receipt of a retransmission order. Analysis

yields [K2]:

$$\frac{T}{D} = \frac{d}{D} + 1 + \frac{1-q}{q_t} \left[\frac{d}{D} + 1 + \frac{K-1}{2} \right] \quad (3-32)$$

$$q = \left[\exp(-G/K) + \frac{G}{K} \exp(-G) \right]^K e^{-S} \quad (3-33)$$

$$q_t = \left[\frac{e^{-G/K} - e^{-G}}{1 - e^{-G}} \right] \left[e^{-G/K} + \frac{G}{K} e^{-G} \right]^{K-1} e^{-S} \quad (3-34)$$

$$\approx \left[(K-1)/K \right] e^{-G}$$

$$S = Gq_t / (1 + q - qt) \quad (3-35)$$

where q and q_t denote the successful transmission probability of newly generated and retransmitted packets, respectively.

As with the simpler case considered initially, T increases with S , slowly at first and then more rapidly as S approaches its maximum allowable value.

Delay T vs throughput S can be plotted for various values of G , K , and d/D . Such plots have been obtained [K2, K5, L4] for a finite number as well as an infinite number of users. For analytical simplicity it is desirable to obtain simpler albeit approximate equations relating the various system parameters. Lam's [L4] data for up to $M=10$ users is closely approximated by the following expression, where $a(M)$ represents degradation of channel capacity due to collisions, $b(M)$ is a scale factor used to fit curves in the vertical direction, d is the link propagation time, and λ the packet generation rate:

$$T = d + \frac{1}{D} + \frac{b(M) \lambda}{2(D a(M) - \lambda)} \cdot \frac{1}{D} \quad (3-36)$$

Fig. 3-6 shows $a(M)$ and $b(M)$ is M , where $a(1) = 1$ and, as $M \rightarrow \infty$ $a(M)$ approaches $1/e$ for slotted ALOAH and $1/2e$ for pure ALOAH as expected.

One can legitimately enquire as to the effects on delay T and throughput S when the usual slotted ALOAH assumptions of infinite user population and identical user message lengths, transmitter power, and traffic rates are relaxed. Some results are available for these more general situations. For pure ALOAH channels, throughput is maximized when all users generate packets of fixed and equal length [A3, F5]. When users generate packets at different rates, however, throughput may increase [A3, K2]. If one user generates most of the traffic his collision probability will be low, while that of the other users will be high. The large user's throughput and therefore the average throughput will be high, with low delay. The throughput and delay of the other users will approach that of an infinite user population. Analysis shows that for an infinite user population, throughput increases and delay decreases when a few users generate most of the traffic [K2, A3].

When transmitter power varies among users overall channel throughput is increased via channel capture. In collisions with low power users, high power users' transmissions are not obliterated. When a low power user encounters collision, he must retransmit. Thus, the high power users' throughput increases and delay decreases, while that for the low power users remains unchanged. Metzner [M6] showed that division of users into two power groups increases overall throughput by approximately 50%.

Number of Users M	a(M)	b(M)
1	1	1
2	0.531	3.059
3	0.528	4.674
5	0.494	5.871
10	0.489	7.219

Fig. 3-6 Parameters $a(M)$ and $b(M)$ [H5].

Abramson [A3] has obtained some throughput results for situations where users are distributed on an annular ring centered on a central transceiver node. As noted earlier, differing receiver power levels among users is, in effect, a priority assignment scheme.

Most of the available results for graded user power involve throughput calculations. Delay vs traffic offered is not as readily available, although derivation of delay equations would follow the approach illustrated at the beginning of this section or in Lam [L4].

Comparisons of pure and slotted ALOAH with FDMA and TDMA show the former to be much better in terms of lower delay for a given throughput except in those instances when traffic offered $G \rightarrow \infty$, in which case ALOAH techniques show vanishingly small throughput and delay which grows without bound [K2, T3]. Performance comparisons with SSMA and other PAMA techniques seem unavailable.

III-6 Modified Contention Multiplexing

Some of the modifications proposed to improve contention multiplexing throughput and delay are considered here.

Metzner [M6] showed that optimum partitioning of users into two power categories increases utilization from $1/e$ (36.8%) to 53% for slotted ALOAH and from $1/2e$ (18.4%) to 26.5% for pure ALOAH. Optimum division of the users into 18 power categories resulted in 90% utilization, assuming that users are not obliterated by collisions with those in lower power categories [M6].

As noted earlier, power imbalance implies user priorities.

A priority-for-fee system might be attractive, although users would not likely split optimally into classes [S9]. To achieve optimal partitions, users might regularly rotate among various power classes in a way consistent with their priority needs. Priority assignment complexities and power ranges required to effect capture by a user over those of lower priority would establish practical limits to the number of priority classes.

When a radio network node is accessed by terminals with equal power but varying distances from a repeater node, terminals r units from the repeater will not be obliterated by those a distance mr ($m > 1$) from the repeater. Abramson [A3] has considered this situation and shows, among other things, that with traffic spread uniformly over an area, throughput equals 0.5 for slotted ALOAH, which exceeds the $1/e$ throughput when all terminals are equidistant from the repeater. Here again power differentials at the site of collision improves throughput.

Another way to improve throughput and delay is to employ user carrier sensing [K2, T3, T4, K6], provided the round trip time is much less than message (or packet) duration. In non-persistent carrier sensed multiple access (CSMA) each terminal (or node) with data to transmit monitors the channel and operates as follows [K2, K6].

1. If the channel is idle, the terminal transmits its packet.
2. If the channel is busy, transmission is rescheduled to some later time in accordance with a delay distribution, and the algorithm is repeated.

Non-persistent CSMA can be operated in either the unslotted or slotted (synchronized) mode. Implicit in the discussion here is the assumption that each terminal can "hear" all others in the

system.

An alternative to non-persistent CSMA is p-persistent CSMA which operates as follows:

1. If the channel is sensed idle, then with probability p the terminal transmits its packet, and with probability $1-p$ delays its transmission d , sec. where d is the link propagation time. If at this later time the channel is idle, the above process is repeated, otherwise transmission is rescheduled according to a delay distribution.
2. If the terminal senses a busy channel, it waits (persists in sensing) until the channel is idle and then operates as 1. and 2.

Throughput has been obtained for unslotted and slotted non-persistent CSMA as well as for p-persistent CSMA [K2, K6, T3] with G as the amount of traffic offered per packet length D and $a = d/D$; throughput S is as follows [K2, T4]:

Nonpersistent CSMA:

$$S = \frac{Ge^{-aG}}{G(1+2a)+e^{-2G}} \quad (3-37)$$

Slotted nonpersistent CSMA:

$$S = \frac{aGe^{-aG}}{1+a-e^{-aG}} \quad (3-38)$$

p-Persistent CSMA:

$$S = \frac{(1-e^{-aG})P_s^1 \Pi_0 + P_s(1-\Pi_0)}{(1-e^{-aG})[a\bar{t}^1 \Pi_0 + a\bar{t}(1-\Pi_0) + 1 + a] + a\Pi_0} \quad (3-39)$$

Constants \bar{t} , \bar{t}^1 , P_s , P_s^1 and Π_0 are defined elsewhere [K2, K6].

Plots of S vs G for $a \ll 1$ show both slotted and pure ALOAH schemes to be inferior to the various CSMA schemes. The persistent schemes show highest throughput for $G < 1$ while the non-persistent schemes are best for $G \gg 1$. For $a \gg 1$, in which case the link propagation time is much greater than the packet length, the ALOAH schemes are best, since any information obtained from channel sensing is ancient history. (See Fig. 3-7).

For the various CSMA modes average packet delays can be calculated by recognizing $(G/S) - 1$ to be the retransmission rate relative to the throughput. Thus, for nonpersistent CSMA [K2, K6]

$$T = \left(\frac{G}{S} - 1 \right) (2a + 1 + B + \bar{X}) + 1 + a \quad (3-40)$$

where \bar{X} is the mean of the uniformly distributed retransmission delay and B is the normalized time to receive an acknowledgement of a successfully transmitted packet.

The above discussion of CSMA assumes that any terminal can hear all the others sharing a common band. If some terminals are hidden from others, then CSMA performance deteriorates. For example, if all N users accessing a central node are divided into two groups of $N/2$ users each, and if all members of each group are hidden from all members of the other group then the maximum throughput of nonpersistent and 1-persistent CSMA schemes falls from 0.82 and 0.53 to 0.29 and 0.27, respectively, which is midway between the maximum throughput for slotted and pure ALOAH.

To combat the hidden terminal problem, busy-tone multiple access (BTMA) may be employed. In BTMA the node being accessed by a group of terminals uses a small portion of the shared channel to

ACCESS SCHEME	MAXIMUM THROUGHPUT
pure ALOAH	0.184
slotted ALOAH	0.368
1-persistent CSMA	0.529
slotted 1-persistent CSMA	0.531
non-persistent BTMA	0.70
0.1-persistent CSMA	0.791
non-persistent CSMA	0.815
0.03-persistent CSMA	0.827
slotted non-persistent CSMA	0.857
perfect scheduling	1.000

Fig. 3-7 Maximum throughput for various multiplexing schemes: $a = 0.01$ [K2].

indicate its unavailability. A terminal with a waiting packet observes the channel for T_d sec., and then decides whether or not the channel is busy; T_d should be optimized [K2, T4].

When $a \gg 1$, a portion $(1-k)W$ of the shared channel bandwidth W can be used for reserving transmission slots and thereby avoiding collisions. In the request answer-to-request (RAM) scheme [T3] the control channel is further subdivided into one request and one answer channel. Terminals with messages to transmit access the request channel using either pure or slotted ALOAH multiplexing. The requests are very short and specify the terminal's identity and the message length. The answer channel, on which contention is absent is used to send the time at which the terminal in question is to transmit. Transmission of the actual message on the main channel occurs without collisions.

The average message delay T and the delay variance is readily obtained by first observing that

$$T = T_1 + T_2 \quad (3-41)$$

where T_1 is the time between request generation and receipt by a central station, and T_2 is the delay between the station's receipt of the request and the completion of transmission of the actual message. For ALOAH multiplexing of requests:

$$T_1 = T_A (S_r) 2v b_m / (1-k)W \quad (3-42)$$

where T_A and S_r denote the ALOAH delay and request input rate, respectively, b_m the average number of bits per message, and v the ratio of the length in bits of the request to the actual message. Equal capacity is assumed for the request and answer channel.

An approach similar to the one above yields T_1 for request transmission on carrier sensed channels [T3].

Similarly,

$$T_2 = \left[\frac{2v}{1-k} + \frac{1}{k} + \frac{\lambda T_m \phi / 2k}{1 - \lambda T_m} \right] \frac{b_m}{W} + 2d \quad (3-43)$$

where λ is the (Poisson) message arrival rate, T_m the average time to transmit a message on the message channel, and ϕ is as defined in Section 3-3. The terms in (3-43), include transmission of the answer to the request, time to transmit the message, queuing delay for the message, and propagation delay of both the message and the answer-to-request.

Delay T depends on k which can be optimized. In plotting delay T vs throughput S , it is seen that as S increases, the reservation scheme is superior to pure and slotted ALOAH as well as to channel sensing schemes. Also the reservation scheme becomes superior to channel sensing as parameter a increases. For low traffic levels, reservations incur unnecessary delay because of the time needed to transmit reservation data. The number of system parameters involved frustrate attempts to briefly summarize performance comparisons; the reader is referred instead to the detailed discussions in the excellent paper by Tobagi and Kleinrock [T3].

Other reservation schemes have been evaluated, including one where the answer-to-request is retained at the station until the terminal initiating the request is to begin message transmission. This latter scheme seems at least as good as the one considered in the above paragraph [T3].

Regarding implementation, pure ALOAH is obviously the simplest, but also least effective in heavy traffic in terms of delay and throughput. Improved efficiencies involve additional costs of timing control, power level control, channel sensing or a combination of these. These costs and complexities tend to increase with reduced collision probabilities which in turn increase throughput and reduce delay. An interesting cost study which compares various satellite multiplexing schemes with each other as well as with terrestrial links is reported by Eric [E1].

Further study of modified contention multiplexing schemes is clearly warranted. Optimization of system parameters, sensitivity of performance to system parameters and traffic statistics and implementation cost considerations are of specific interest. Also of interest is the approximation of delay-vs-throughput curves [K2, K6, T3, T4] for the various multiplexing schemes by relatively simple equations like (3-36), to facilitate network design. An approximation like (3-36) seems feasible, since delay-vs-throughput curves for modified contention schemes lie between curves for unslotted ALOAH and perfect scheduling which involves delays due solely to queuing.

III-7 Contention Multiplexing and Queuing Interactions

Results from preceding sections combine to yield delays which occur when queues form at nodes which use contention accessing of a shared channel. Queuing delays were excluded from analyses in the preceding two sections by assuming that any contending node had at most one message awaiting access. (The reservation scheme analysis considered queuing of the messages

but not the reservations.)

Assume initially that each message must be acknowledged as received before the next message can be transmitted. The mean service time \bar{X} must then include delays which result from retransmissions. These occur frequently near maximum throughputs, and in slotted and pure ALOAH accessing retransmission traffic constitutes 63% and 72% of the total traffic, respectively. Thus

$$\bar{X} = \frac{1}{\mu R} + d + Q \quad (3-44)$$

$$\bar{X}^2 = \left(\frac{1}{\mu R} + d\right)^2 + 2Q \left(\frac{1}{\mu R} + d\right) + Q^2 \quad (3-45)$$

where μ is the average message length, R the transmission rate in bits, d the link propagation time and Q the average waiting time due to retransmissions, as described in Section 3-5.

The total delay, including queuing delay W plus service time \bar{X} is

$$\begin{aligned} T &= W + \bar{X} \\ &= \lambda \bar{X}^2 \phi / 2(1-\rho) + \bar{X} \end{aligned} \quad (3-46)$$

where $\rho = \lambda \bar{X}$ and λ is the (Poisson) arrival rate of messages at the node. Determination of ϕ requires knowledge of the distribution of \bar{X} . A Gamma distribution with parameters selected to yield observed \bar{X} and \bar{X}^2 would likely yield reasonably accurate values of T [M4].

The above mode of operation would be used in CSMA and BTMA schemes where $d \ll 1/\mu R$ and might be used in satellite applications, even though $d \gg 1/\mu R$. However another mode of operation would

likely be more appropriate for satellite channels. Messages in the queue would be transmitted one after the other, without awaiting acknowledgements of successful transmissions. If an acknowledgement to a particular message is not received after a time-out interval then the message is assumed lost, and rejoins the queue on a non-priority basis. For this mode of operation

$$\bar{X} = 1/\mu R \quad (3-47)$$

and T is given by (3-46), except that λ now includes both new and retransmitted packets.

The value of λ to be used in (3-46) can be determined from $S = S(G)$, where S/G is the probability that an attempt at transmission is successful. If λ_n denotes the arrival rate of new messages at any node, $\lambda = G\lambda_n/S$, where S and G are defined in Section 3-5 and

$$\rho = (G/S)(\lambda_n/\mu R) \quad (3-48)$$

Which of the two schemes is best? The question is a difficult one, and is considered in Chapter 5 in dealing with data link errors and flow control.

In some situations, for reasons of flow control retransmissions would have priority over new messages, in which case the delay for the second scheme would require inclusion of a priority structure in calculation of the queuing delay (see Section 5-4).

An approach like the one presented above enables queuing and retransmission delays to be combined to obtain total delay for other contention and modified contention accessing schemes. The delay for the entire user population is calculated using

$T = \sum_{i=1}^M \lambda_i T_i$, where M includes all network links as explained in Section 2-4.

Further studies on modified contention multiplexing suggested at the conclusion of Section 3-6 should include interaction between queuing and retransmissions resulting from collisions.

III-8 Polling

Roll-call polling prevents collisions between messages sharing a common channel. One central node successively polls the access nodes; a node holding messages sends some or all of these to the polling centre, after which the next user is polled.

Time elapses while a node awaits polling during which time additional messages may enter the nodal buffer. Further delay occurs during actual message transmission. In transmitting messages from the polling centre delays occur because only one of the M users can be served at any one time.

Mean delays for a group of M users sharing a polled channel system have been determined [T3, K7]. For M users the waiting time W is:

$$\frac{W}{\tau} = \frac{1}{2} \cdot \frac{M\tau}{1-M\bar{X}} + \frac{1-\bar{X}}{2} + \frac{1}{2} \cdot \frac{Mr(1-\bar{X})}{1-M\bar{X}} \quad (3-49)$$

where \bar{X} and τ denote, respectively, the mean and variance of the number of τ -sec. slots required to service messages arriving at a nodal buffer during a τ -sec. interval, and r denotes the number of τ -sec. slots required for switching to the next node [T3]. The normalized time delay equals W plus the packet transmission time.

To illustrate use of (3-45) assume that all transmitted data consists either of b_p -bit polling packets or b_m -bit message packets.

Typically $10 \lesssim b_m/b_p \lesssim 100$. Let d be the propagation delay between any terminal and the polling station, and T_p and T_m be the transmission time of a packet and message, respectively. Define $a = d/T_m$ and $b = T_p/d$. In a packet radio environment $a \approx 0.01$.

With $\tau = d$ one time slot is required for the polling packet to reach the terminal being polled, and one additional time slot must elapse before the polling station can decide whether to allocate the polled terminal or the transmission channel. Thus,

$$r = b + 2 \quad (3-50)$$

If $L = b_m/b_p = 100$, then $b = 1$ and $r = 3$. If $L = 10$, $b = 10$ and $r = 12$.

If packets arrive at each terminal with a Poisson distribution with mean rate λ then

$$\begin{aligned} \bar{X} &= \lambda T_m / d \\ &= \lambda / a \end{aligned} \quad (3-51)$$

$$\tau^2 = \lambda / a^2 \quad (3-52)$$

Substitution of (3-50) - (3-52) into (3-49) yields W .

When $b < 1$ it is sometimes convenient to define slot size $\tau = T_p$. In this case one slot equals a polling packet, and

$$\bar{X} = \lambda L \quad (3-53)$$

$$\sigma^2 = \lambda L^2 \quad (3-54)$$

It is seen from (3-49) that when $\bar{X} \rightarrow 0$ and $\sigma \rightarrow 0$, $W = (Mr+1)/2$. In this case, waiting delay is due solely to the delay experienced

by a terminal awaiting polling. In this situation, polling is less desirable than contention accessing, since collisions would rarely occur. However, when channel utilization is heavy collisions are commonplace, and polling prevents such collisions, although delays do occur while any individual terminal waits for the opportunity to transmit. Polling is useful when inherent delays are tolerable or when the number of users M is not too large. Polling delays are less than those of non-persistent CSMA and BTMA for ρ above a certain value $[T3]$.

The preceding analysis assumes message flow from terminal nodes to polling station. When flow is reversed $r = 0$, since the central location need suffer no polling delay.

In some situations hub polling $[S5, S8]$ is used to reduce the time which elapses while the central station interrogates each node. As (3-49) shows, this time is a larger contribution to delay under light channel utilization, since many terminals must be polled, on the average, before one having a waiting message is encountered. In hub polling, a node completes its transmission and then sends directly to the next node on the polling list a signal to begin transmission. Hub polling requires that terminals be linked directly in loop fashion, and such a linking implies additional data channels and cost which may not be warranted if the nodes are widely separated geographically.

Pack and Whitaker $[P3]$ have compared polling with contention access for an on-line credit verification application. The comparison is instructive, even though most of the system parameters are fixed at the outset and not subjected to optimization.

III-9 References

- A1 J.M. Aein and O.S. Kosovych, "Satellite capacity allocation," Proc. IEEE, vol. 65, pp. 332-342, March 1977.
- A2 N. Abramson and F.F. Kuo, Eds., Computer-Communication Networks. Englewood Cliffs, N.J.: Prentice-Hall, 1973.
- A3 N. Abramson, "The throughput of packet broadcasting channels," IEEE Trans. Commun., vol. COM-25, pp. 117-128, Jan. 1977.
- B1 T. Berger and D.W. Tufts, "Optimum pulse amplitude modulation, Part I: Transmitter-receiver design and bounds from information theory," IEEE Trans. Inform. Theory, vol. IT-13, pp. 196-208, Apr. 1967.
- C1 A. Cantoni and P. Butler, "Linear mean-square error estimators applied to channel estimators," IEEE Trans. Commun., vol. COM-25, pp. 441-446, Apr. 1977.
- C2 A. Cantoni and P. Butler, "Stability of decision feedback inverses," IEEE Trans. Commun., vol. COM-24, pp. 970-977, Sept. 1976.
- C3 J.K. Cavers, "Variable rate transmission for Rayleigh fading channels," IEEE Trans. Commun., vol. COM-20, pp. 15-22, Feb. 1972.
- C4 J.K. Cavers and S.K. Lee, "A simple buffer control for variable-rate communication systems," IEEE Trans. Commun., vol. COM-24, pp. 1045-1048, Sept. 1976.
- D1 J. R. Davey, "Modems," Proc. IEEE, vol. 60, pp. 1284-1292, Nov. 1972.
- E1 M.J. Eric, "The economics and performance of satellite packet switching," IEEE Trans. Commun., vol. COM-23, pp. 732-741, July, 1975.
- F1 D.D. Falconer and F.R. Magee, Jr., "Evaluation of decision feedback equalization and Viterbi algorithm detection for voice-band data transmission - Part I," IEEE Trans. Commun., vol. COM-24, pp. 1130-1139, Oct. 1976.
- F2 D.D. Falconer and F.R. Magee, "Evaluation of decision feedback equalization and Viterbi algorithm detection for voice-band data transmission - Part II," IEEE Trans. Commun., vol. COM-24, pp. 1238-1245, Nov. 1976.
- F3 S.A. Fredricson, "Joint optimization of transmitter and receiver filters in digital PAM systems with a Viterbi detector," IEEE Trans. Inform. Theory, vol. IT-22, pp. 200-209, Mar. 1976.
- F4 G.D. Forney, "The Viterbi algorithm," Proc. IEEE, vol. 61, pp. 268-278, Mar. 1973.

- F5 M.J. Ferguson, "A bound and approximation of delay distribution for fixed-length packets in an unslotted ALOAH channel and a comparison with time division multiplexing (TDM)," IEEE Trans. Commun., vol. COM-25, pp. 136-140, Jan. 1977.
- G1 W.A. Gardner, "An equivalent linear model for marked and filtered doubly stochastic Poisson processes with application to MMSE linear estimation for synchronous m-ary optical data signals," IEEE Trans. Commun., vol. COM-24, pp. 917-921, Aug. 1976.
- H1 F.S. Hill, Jr., "A unified approach to pulse design in data transmission," IEEE Trans. Commun., vol. COM-25, pp. 346-354, Mar. 1977.
- H2 E.V. Hoverstein, D.L. Snyder, R.O. Harger and K. Kurimoto, "Direct-detection optical communication receivers," IEEE Trans. Commun., vol. COM-22, pp. 17-27, Jan. 1974.
- H3 P.M. Hopkins and K.S. Simpson, "Probability of error in pseudo-noise (PN) - modulated spread spectrum binary communication systems," IEEE Trans. Commun., vol. COM-23, pp. 467-472, April, 1975.
- H4 J.F. Hayes, "Adaptive feedback communications," IEEE Trans. Commun. Technol., vol. COM-16, pp. 29-34, Feb. 1968.
- H5 D. Huynh, H. Kobayashi, and F.F. Kuo, "Optimal design of mixed-media packet-switching networks: routing and capacity assignment," IEEE Trans. Commun., vol. COM-25, pp. 158-169, Jan. 1977.
- K1 L. Kleinrock, Queueing Systems, Vol. 1: Theory. New York, N.Y.: Wiley, 1975.
- K2 L. Kleinrock, Queueing Systems, Vol. 2: Computer Applications. New York, N.Y.: 1976.
- K3 H. Kobayashi and A.G. Konheim, "Queueing models for computer communications system analysis," IEEE Trans. Commun., vol. COM-25, pp. 2-29, Jan. 1977.
- K4 A.G. Konheim, "Chaining in a loop system," IEEE Trans. Commun., vol. COM-24, pp. 203-210, Feb. 1976.
- K5 L. Kleinrock and S.S. Lam, "Packet switching in a multi-access broadcast channel: performance evaluation," IEEE Trans. Commun., vol. COM-23, pp. 410-423, Apr. 1975.
- K6 L. Kleinrock and F.A. Tobagi, "Packet switching in radio channels: part I - carrier sense multiple-access modes and their throughput-delay characteristics," IEEE Trans. Commun., vol., COM-23, pp. 1400-1417, Dec. 1975.
- K7 A.G. Konheim and B. Meister, "Waiting lines and times in a system with polling," IBM J. Watson Research Centre, Yorktown Heights, N.Y.: Rept. RC 3841, May 1972.

- L1 R.W. Lucky, J. Salz and E.J. Weldon, Jr., Principles of Data Communication. New York, N.Y.: McGraw-Hill, 1968.
- L2 W.C. Lindsay and M.K. Simon, Telecommunication Systems Engineering. Englewood Cliffs, N.J.: Prentice-Hall, 1973.
- L3 S. Lin, An Introduction to Error-Correcting Codes. Englewood Cliffs, N.J.: Prentice-Hall, 1973.
- L4 S.S. Lam and L. Kleinrock, "Packet switching in a multi-access broadcast channel: dynamic control procedures," IEEE Trans. Commun., vol. COM-23, pp. 891-905, Sept. 1975.
- L5 K.D. Levin, "The overlapping problem and performance degradation of mobile digital communication systems," IEEE Trans. Commun., vol. COM-23, pp. 1342-1347, Nov. 1975.
- M1 F.R. Magee, Jr., "A comparison of compromise Viterbi algorithm and standard equalization techniques over bandlimited channels," IEEE Trans. Commun., vol. COM-23, pp. 361-367, Mar. 1975.
- M2 K.H. Mueller and M. Miller, "Timing recovery in digital data receivers," IEEE Trans. Commun., vol. COM-24, pp. 516-531, May 1976.
- M3 T.V. Muoi and J.L. Hullet, "Receiver design for multilevel digital optical fibre systems," IEEE Trans. Commun., vol. COM-23, pp. 987-994, Sept. 1975.
- M4 J. Martin, Systems Analysis for Data Transmission. Englewood Cliffs, N.J.: Prentice-Hall, 1972.
- M5 S.A. Mamoud, "Resource Allocation and File Access Control in Distributed Information Networks," Carleton Univ. Dept. of Systems Enggr. Tech. Rept., Carleton University, Ottawa, Canada, Jan. 1975.
- M6 J.J. Metzner, "On improving utilization in ALOAH networks," IEEE Trans. Commun., vol. COM-24, pp. 447-448, Apr. 1976.
- P1 S.D. Personick, "Receiver design for digital fiber optic communication systems: Parts I and II," Bell Syst. Tech. J., vol. 52, pp. 843-886, July-Aug. 1973.
- P2 A. Papoulis, Probability, Random Variables and Stochastic Processes. New York, N.Y.: McGraw-Hill, 1965.
- P3 C.D. Pack and B.A. Whitaker, "Multipoint private line (MPL) access delay under several interstation disciplines," IEEE Trans. Commun., vol. COM-24, pp. 339-348, March 1976.
- Q1 S.V.H. Qureshi, "Timing recovery for equalized partial-response systems," IEEE Trans. Commun., vol. COM-24, pp. 1326-1331, Dec. 1976.

- R1 M.P. Ristenbatt, "Alternatives in digital communications," Proc. IEEE, vol. 61, pp. 703-721, June 1973.
- R2 P.K. Runge, "An experimental 50 Mb/s fiber optic PCM repeater," IEEE Trans. Commun., vol. COM-24, pp. 413-418, Apr. 1976.
- R3 I. Rubin, "An approximate time-delay analysis for packet-switching communication networks," IEEE Trans. Commun., vol. COM-24, pp. 210-222, Feb. 1976.
- S1 J. Salz, "Optimum mean-square decision feedback equalization," Bell Syst. Tech. J., vol. 52, pp. 1341-1374, Oct. 1973.
- S2 J.J. Stiffler, Theory of Synchronous Communication. Englewood Cliffs, N.J.: Prentice-Hall, 1971.
- S3 M.K. Simon and W.C. Lindsey, "Optimum performance of suppressed carrier receivers with Costas loop tracking," IEEE Trans Commun., vol. COM-25, pp. 215-227, Feb. 1977.
- S4 R. Syski, Introduction to Congestion in Telephone Systems. London, Eng.: Oliver and Boyd, 1960.
- S5 M. Schwartz, Computer-Communication Network Design and Analysis. Englewood Cliffs, N.J.: Prentice-Hall, 1977.
- S6 T. Saaty, Elements of Queueing Theory. New York, N.Y.: McGraw-Hill, 1961.
- S7 J.W. Schwartz, J.M. Aein and J. Kaiser, "Modulation techniques for multiple access to a hard-limiting satellite repeater," Proc. IEEE, vol. 54, pp. 763-776, May 1966.
- S8 J.W. Schwartz and M. Munter, "Multiple-access communications for computer nets," in [A2], Ch. 8.
- S9 M. Schwartz, R.R. Boorstyn and R.L. Pickholtz, "Terminal oriented computer networks," Proc. IEEE, vol. 60, pp. 1415-1418, Nov. 1972.
- T1 D. Tufts and T. Berger, "Optimum pulse amplitude modulation, Part II: Inclusion of timing jitter," IEEE Trans. Inform. Theory, vol. IT-13, pp. 209-216, April 1967.
- T2 Y. Tkasaki, M. Tanaka, N. Maeda, K. Yamashita and K. Nagano, "Optical pulse formats for fiber optic digital communications," IEEE Trans. Commun., vol. COM-24, pp. 404-413, Apr. 1976.
- T3 F.A. Tobagi and L. Kleinrock, "Packet switching in radio channels: Part III - polling and (dynamic) split-channel reservation multiple access," IEEE Trans. Commun., vol. COM-24, pp. 832-845, Aug. 1976.
- T4 F.A. Tobagi and L. Kleinrock, "Packet switching in radio channels: Part II - the hidden terminal problem in carrier sense multiple-access and the busy tone solution," IEEE Trans. Commun., vol. COM-23, pp. 1417-1434, Dec. 1975.

- U1 G. Ungerboeck, "Fractional tap-spacing equalizer and consequences for clock recovery in data modems," IEEE Trans. Commun., vol. COM-24, pp. 856-864, Aug. 1976.
- V1 P.J. Van Gerwin, N.A.M. Verhoeckx, H.A. van Essen, and F.A.M. Snijders, "Microprocessor implementation of high-speed data modems," IEEE Trans. Commun., vol. COM-25, pp. 238-250, Feb. 1977.
- W1 J.M. Wozencroft and I.M. Jacobs, Principles of Communication Engineering, New York, N.Y.: Wiley, 1965.

IV NETWORK DESIGN

IV-1 Network Design: Variables and Constraints

The network design problem involves optimization of the network topology, link capacities and link flows to minimize either cost, delay or reliability with constraints on the other two quantities. In some situations one or more of the design variables may be specified, in which case optimization is with respect to the others. As noted in Chapter 2, a formal reliability constraint is often omitted, although the degree of network connectivity may be constrained.

Sections 4-2, 4-3 and 4-4 deal, respectively, with optimization of link capacities, optimization of link flows, and joint optimization of link flows and capacities. Sections 4-5, 4-6 and 4-7 deal with topological optimization of centralized networks.

Sections 4-8, 4-9 and 4-10 consider topological design of distributed networks. Section 4-8 describes methods that avoid initial clustering of nodes. Section 4-9 considers backbone node location which arises from subdivision of the problem into several simpler problems involving local access network design and distributed network design. Section 4-10 considers network design via clustering, and relates this to the problem of routing in large networks.

Network cost vs traffic level is considered within the framework of economies of scale in Section 4-11. Unresolved issues are discussed in Section 4-12.

IV-2 Data Link Capacity Optimization

Optimization of data link capacity is straightforward in principle. Network topology and link flows are specified, and so, therefore, is the reliability U as defined in Section 2-4, provided the link reliabilities are independent of link capacities C_i . It remains to select the capacity C_i of each data link to minimize the delay T subject to a constraint on the total cost D , or equivalently to minimize D subject to a constraint on T .

Two difficulties arise in practise. First, actual cost-vs-capacity functions are discreet and discontinuous rather than well-behaved and continuous functions, as explained in Section 2-3. Second, the algebraic expressions for T and D in terms of the system capacities may be complex even if actual cost-vs-capacity relationships are approximated by well-behaved functions.

For centralized networks an algorithm exists for minimizing the cost D subject to a constraint on the average delay $[F1, F2]$. For distributed networks either of the following two sub-optimization algorithms may be used if the number of discreet capacities or data links is not too large $[G1, G2]$:

A. Bottom Up Algorithm

1. Assign the minimum available capacities to each link
2. Calculate network cost D
3. If D exceeds the maximum allowable cost D_M , stop and accept as optimum the existing capacities. Otherwise increment to the next available capacity the capacity of that link whose utilization factor ρ is largest and repeat steps 2 and 3.

B. Top Down Algorithm

1. Assign the maximum available capacities to each link
2. Calculate network cost D
3. If $D < D_M$ stop; capacities resulting from the last iteration are accepted as optimum. Otherwise decrease to the next available capacity the capacity of the link for which ρ is smallest and repeat steps 2 and 3.

The above algorithms can be modified in an obvious way to optimize T rather than D . If reliability U depends on capacities C_i through failure probabilities p_i in (2-13) U can be checked at each iteration. Capacity assignments which violate reliability constraints would be discarded.

The above procedure seems to generate capacity assignments which are either optimum or close to optimum in most cases, particularly when economies of scale as regards capacity costs are modest $[G1, G2]$.

When the number of discrete capacities is large, stepwise cost-vs-capacity curves may be approximated by continuous functions. Use of continuous functions has the additional advantage of providing analytical insight to the dependence of network behaviour on data-link capacities. Capacities so obtained would be replaced by slightly larger discrete values when actually realizing the network.

Consider a network with M point-to-point data links of capacity C_i . Further, assume:

1. independent Poisson message arrivals at each node
2. infinite nodal storage
3. identical message length distributions

4. message service which is independent of message length
5. error-free channels
6. fixed routing of messages.

Recall that the average message delay

$$T = \sum_{i=1}^M (\lambda_i / \gamma) T_i \quad (4-1)$$

where λ_i and T_i denote, respectively, the message arrival rate and delay on the i^{th} channel, and γ is the throughput. Delay T_i is given by (3-20). Define $d_i(C_i)$ as the cost of the i^{th} data channel of capacity C_i ($i=1,2,\dots,M$). Total data channel cost D , which may include termination costs is as follows:

$$D = \sum_{i=1}^M d_i(C_i) \quad (4-2)$$

Minimization of T involves minimization of

$$G = T + A \left[\sum_{i=1}^M d_i(C_i) - D \right] \quad (4-3)$$

with respect to C_i ($i=1,2,\dots,M$) where Lagrange multiplier A is selected to satisfy the cost constraint (4-2).

Differentiation of (4-3) with respect to C_i yields M equations which are decoupled with respect to the capacities. Such decoupling simplifies solution enormously and permits solution of each C_i in terms of multiplier A , which is then obtained from (4-2). The decoupling is due to the message arrival independence assumption on which (3-20) is based [K1, K2].

If nodal processing time $P_i=0$ and propagation delay $K_i=0$ in (3-20), if $\mu_i=\mu_i^{-1}$ (average message length equals average data plus control packet length) and if d_i is linear in C_i then the optimum value of C_i is as follows [K1, K2]:

$$C_i = \frac{\lambda_i}{\mu} + \frac{D_e}{d_i} \frac{\sqrt{\lambda_i d_i}}{\sum_{j=1}^M \lambda_j d_j} \quad (4-4)$$

where "excess cost" D_e is defined as follows:

$$D_e = D - \sum_{i=1}^M \lambda_i d_i / \mu_i \quad (4-5)$$

The resulting delay is

$$T = \frac{\bar{n}}{\mu D_e} \sum_{i=1}^M \sqrt{\left(\frac{\lambda_i d_i}{\lambda} \right)^2} \quad (4-6)$$

where the average path length \bar{n} is defined as follows

$$\begin{aligned} \bar{n} &= \sum_{i=1}^M \lambda_i / \gamma \\ &= \lambda / \gamma \end{aligned} \quad (4-7)$$

If $d_i = d=1$ for all $i=1,2,\dots, M$ then the above equations simplify further:

$$D = \sum_{i=1}^M C_i \quad (4-8)$$

$$D_e = D - \sum_{i=1}^M (\lambda_i / \mu) \quad (4-9)$$

$$C_i = \frac{\lambda_i}{\mu} + \frac{C (1 - \bar{n}\rho) \sqrt{\lambda_i}}{\sum_{i=1}^M \sqrt{\lambda_i}} \quad (4-10)$$

$$T = \frac{\bar{n} \sum_{i=1}^M \sqrt{\lambda_i / \lambda}}{\mu C (1 - \bar{n}\rho)} \quad (4-11)$$

where

$$\rho = \gamma / \mu C \quad (4-12)$$

With $d_i = 1$ for all i , $C=D$ where C is to total capacity available for the network.

It is seen from the above equations that after sufficient capacity is assigned to handle the traffic on each link (i.e. $\lambda_i = \mu C_i$), "excess" capacity is then assigned in proportion to $\sqrt{\lambda_i}$.

As noted in Section 2-3, costs tend to increase more slowly with capacity than linearly. If one assumes $d_i = d_i C_i^\alpha$ where $0 < \alpha \ll 1$, one finds that the optimum C_i solves the following equations [K1]:

$$C_i - (\lambda_i / \mu) - g_i C_i^{(1-\alpha)/2} = 0 \quad (4-13)$$

where

$$g_i = (\lambda_i / \mu \gamma \alpha A d_i)^{1/2} \quad (4-14)$$

Solution of (4-13) is readily accomplished using standard

iteration techniques.

Optimum values of C_i would ultimately be replaced by discrete values. The sensitivity of T and D to changes in C_i are readily obtained from the defining equation (4-1) and (3-20). If $C_i \approx \lambda_i / \mu$, small changes in C_i cause large changes in T_i , with the result that in such cases use of a value of C_i slightly larger than optimum would reduce T_i considerably.

The preceding discussion is based on the minimization of delay subject to a constraint on total cost D . If D is minimized subject to a constraint on T , the optimum capacities and resulting cost for the case $d_i = d_{i0} + d_i C_i$ are as follows [G2]:

$$C_i = \lambda_i + \frac{\sum_{j=1}^M \sqrt{d_j \lambda_j}}{\gamma T} \sqrt{\frac{\lambda_i}{d_i}} \quad (4-15)$$

$$D = \sum_{i=1}^M \left[d_i \lambda_i + d_{i0} + \frac{\left(\sum_{j=1}^M \sqrt{d_j \lambda_j} \right)^2}{\gamma T_M} \right] \quad (4-16)$$

where T_M is the maximum allowable delay.

Since D would normally decrease as T increases, either can be optimized with the other as a constraint. The resulting D vs T contour would be the same in either case, as would the resulting optimum values of C_i .

The above analysis assumes that all communication between nodes is via point-to-point links. Few results are available for capacity optimization when some or all are radio or satellite links

which involve channel sharing using contention or modified contention multiplexing. One recent study [H1] considers capacity optimization of networks which include a satellite repeater as well as point-to-point links. The approach used involved approximation of delay vs throughput curves for satellite channels by (3-36) followed by further simplifying assumptions to decouple the $M+1$ equations obtained by setting $dT/dC_i=0$ ($i=1,2,\dots,M$) and $dT/dC_s=0$; C_s is the satellite link capacity. A linear cost constraint is assumed; thus

$$D = \sum_{i=1}^M d_i C_i + d_s C_s \quad (4-17)$$

where d_s pertains to the satellite link cost. The average delay

$$T = \frac{1}{Y} \left[\sum_{i=1}^M \lambda_i T_i + \lambda_s T_s \right] \quad (4-18)$$

where λ_i and T_i are for the point-to-point portion of the network, and Y includes both point-to-point and satellite traffic. The approximation used to simplify the equations specifying the optimum capacities is to subtract

$$\Delta = \left(1 - \frac{b(M)}{2}\right) \frac{\lambda_s}{Y \mu_s C_s} \quad (4-19)$$

from (4-18); in satellite applications $\Delta \ll T$.

Using the above approximation, optimum values for the capacities are obtained, as follows:

$$C_i = \frac{\lambda_i}{\mu} + \frac{D_e}{d_i} \frac{\sqrt{f_i}}{\sum_{i=1}^M \sqrt{f_i} + \sqrt{f_s/2}} \quad (i=1,2,\dots,M) \quad (4-20)$$

$$f_i = d_i \lambda_i / \mu \gamma \quad (i=1,2,\dots,M) \quad (4-21)$$

$$f_s = d_s \lambda_s / \mu_s \gamma \quad (4-22)$$

$$D_e = D - \left(\sum_{i=1}^M (d_i \lambda_i / \mu) + (d_s \lambda_s / \mu_s a(M)) \right) \quad (4-23)$$

The resulting minimum delay T is as follows:

$$T = \frac{\lambda_s \tau}{\gamma} + \frac{\left(\sum_{i=1}^M \sqrt{f_i} + \sqrt{f_s/2} \right)^2}{D_e} \quad (4-24)$$

where μ_s equals average satellite message length and $\tau \approx 0.26$ sec.

The structure of the equations is evidently similar to those for networks consisting solely of point-to-point links. In particular the comments pertaining to assigning of "excess" capacity in proportion to the product of "excess" cost and $\sqrt{f_i}$ apply. In fact if $\mu = \mu_s$, $d_1 = d_2 = \dots = d_M = d_s$ then the two sets of equations are identical provided the satellite channel is considered as a single channel with total traffic $\lambda_{M+1} = \lambda_s / a(M)$, and $f_{M+1} = f_s / 2$.

The above study [H1] seems to ignore the effects of queueing

delays on the satellite channel. Queueing delays could be included with delays due to retransmissions caused by collisions. It would be useful to determine whether or not the resulting delay-vs-throughput curves for queueing with slotted ALOAH and other shared channel access schemes described in Chapter 3 could be fitted by (3-36) with appropriate values for $a(M)$ and $b(M)$.

The study referred to above [H1] considered a second scheme, called MASTER, in which packets suffering satellite channel collisions were retransmitted via the point-to-point network, thus avoiding repeated collisions. MASTER exhibited reduced delay in those circumstances where satellite costs were high relative to point-to-point channel costs and the total traffic γ was relatively low. Detailed results appear in Huynh's paper [H1].

Further studies on the optimization and performance evaluation of networks which use broadcast as well as point-to-point links is warranted. As indicated earlier, one aspect of such a study would involve approximation of delay-vs-throughput characteristics to facilitate analysis and optimization.

IV-3 Traffic Flow Assignment (Fixed Routing)

Traffic flow assignment involves selection of the M link flows λ_i to minimize delay T , subject to the constraints

$$\lambda_i \leq \mu C_i \quad (i=1,2,\dots,M) \quad (4-25a)$$

$$\{\lambda_i\} \text{ is a multiple commodity flow} \quad (4-25b)$$

satisfying the traffic matrix $[r]$.

The first constraint requires that the flow λ_i on each channel be such that infinite queues be avoided. The second constraint conserves flow at each network node, including traffic origination and destination nodes.

Following our discussions concerning reliability, we would recommend introduction of a third constraint, namely

$$U = \sum_{i=1}^M p_i \lambda_i \leq U_0 \quad (4-26)$$

Node reliabilities may be included in failure probabilities p_i as explained in Section 2-4.

Cost, delay and reliability averaged over the network depend only on the link flows λ_i , and not directly on the way in which the multicommodity flows r_{ij} contribute to these link flows. Thus, different routing of the individual commodities may yield the same average delay, cost and reliability.

The flow assignment problem arises only for multiconnected networks, in which case traffic from at least one source node i can flow to at least one destination node j along two or more paths. In singly-connected networks there is a single unique path for each source-destination traffic flow, with the result that the flow assignment problem as such does not exist.

One can regard T as a multidimensional surface in $M+1$ dimensional space, with $\lambda_1, \lambda_2, \dots, \lambda_M$ as the (constrained) independent variables and T as the dependent variable. The required non-

negativity of $\{\lambda_i\}$ as well as capacity and possibly reliability constraints further restrict $\{\lambda_i\}$ to a finite region of the first quadrant of the $\lambda_1, \lambda_2, \dots, \lambda_M$ hyperplane. Flow conservation at each node further restricts $\{\lambda_i\}$.

It is not difficult to show that $\partial T / \partial \lambda_i > 0$ and $\partial^2 T / \partial \lambda_i^2 > 0$, where T is defined in (4-1) and (3-20) so long as (4-24) is satisfied and nodal processing delay K_i in (3-20) is independent of $\{\lambda_i\}$. It then follows that T is a convex function of the flows λ_i , with the result that a unique set of $\{\lambda_i\}$ exists to minimize T .

Determination of the optimal flows λ_i and the fractions of λ_i allotted to traffic with source node i and destination node j (the multicommodity flows) can be solved using existing techniques for multicommodity flow problems [C1,D1,H2]. Unfortunately, these techniques are computationally cumbersome to the point that they really can't be used in situations where flows must be optimized hundreds of times in applications involving not only optimization of $\{\lambda_i\}$ but also $\{C_i\}$ and topology. For this reason, computationally efficient or heuristic routing algorithms have been developed.

Most flow assignment algorithms select initial values for flows and iteratively update these until further changes result in delay reductions which fall below a threshold. Selection of the initial "guess" is in itself an important problem, and has some bearing on the speed of solution [S1,F3,K1].

We do not consider here the details of the various available flow optimization algorithms. The flow deviation (FD) algorithm [G1,K1,F4] is computationally efficient and provides an exact (with stopping tolerances) optimum solution. The extremal flow (EF) algo-

rithm [C1] is also reasonably efficient and is also "exact", and can be extended to handle different classes (priorities) of messages. The quadiant projection (GP) algorithm [S2] is again "exact" and provides the routing of the individual multicommodity flows r_{ij} , rather than the global flows $\{\lambda_i\}$, although it could be modified to provide only these, if desired. This latter algorithm seems primarily suited to routing determination in small networks or in those for which only a subset of the nodes communicate with one another. The GP algorithm's rate of convergence normally exceeds that of the FD algorithm, but its complexity is greater.

Following determination of the global flows λ_i , the multicommodity flows are determined and routing tables assigned to each node. The routing table at node i consists of Nl_i entries, where N and l_i denote, respectively, the number of destination nodes and nodes linked directly to node i . Each entry specifies the fraction of traffic ultimately destined for node j that is to be routed through node i . For large networks having high connectivity, the routing tables will be so large that efficient network operation will be jeopardized unless design procedures are modified to substantially reduce table length. This important matter is considered in Section 4-10.

As they currently exist, flow assignment algorithms do not incorporate a reliability constraint. Modification of the FD, EF and GP algorithms would not seem to create substantial difficulties. A study of the effects of such modifications on both the algorithms and the resulting networks would be of considerable interest.

Our discussion has presumed static flows which do not change

once specified. Adaptive routing, which involves changing $\{\lambda_i\}$ in response to changes in traffic and network conditions, is considered in Chapter 5.

IV-4 Flow and Capacity Assignment

The traffic flow and capacity assignment problem involves joint optimization of the flows $\{\lambda_i\}$ and the capacities $\{C_i\}$. Normally, network cost is minimized, subject to a constraint on the average delay T . Additional constraints are those listed in the previous section for flow assignment. We would also advocate a formal reliability constraint, as discussed earlier.

Cost D as a function of $\{\lambda_i\}$ and $\{C_i\}$ consists of many local minima [K1]. Global minimization involves selection of various initial starting values for flows, followed by joint optimization of capacities and flows. If starting flows are randomly selected, it is often conjectured with apparent justification [G2] that, of the resulting collection of locally optimum networks, the one with the lowest cost will be close to optimum.

When link capacities show a linear cost-vs-capacity relationship, cost can be expressed in terms of flows, according to (4-16). The resulting cost-vs-flows equation can then be optimized using flow deviation, assuming no reliability constraint. If cost-vs-capacity costs show a continuous power law behaviour, joint optimization can again be implemented using (4-13), (4-14) and the FD algorithm since cost is again a concave function of $\{\lambda_i\}$ [G1,G2]. For a discrete cost-vs-capacity schedule, iterative application of the flow deviation algorithm preceded by either the top down or bottom up algorithm for capacity optimization (see Section 4-2)

would be employed [G2,G1].

What is the difference in the networks and costs generated by the various algorithms? In general, we don't know. However, in using the FD algorithm with bottom up capacity optimization, and with top down capacity optimization, and in using FD with a power law approximations to cost-vs-data link capacities, the three methods generated "optimum" 26-node ARPA networks whose costs differed by less than 3%. The latter two methods yielded identical networks. Execution time for each algorithm was approximately 90 sec. on an IBM 360/g1. computer.

The comments in the previous two sections regarding inclusion of the formal reliability constraint (2-13) apply here. The flow optimization procedure would be modified to include the constraint, while the constraint would affect the capacity optimization phase only if the failure probabilities were capacity dependent.

IV-5 Topological Design of Centralized Networks Using Graph Heuristics

A centralized network consists of a number of terminals whose geographic location and average traffic rate is assumed known. All traffic flow is between the terminals and the central node which may consist of a CPU housing data bases or computational facilities, or a backbone or gateway node which is part of a large distributed high speed data network. The centralized network design problem involves specification of the following [B1,D2] :

1. The number, location, capacity and type of generic access facilities (GAF) which may be a data concentration, multiplexer or polling centre.

2. The capacity and topological layout of the data links.

Normally, centralized networks consist of a single unique path from each terminal to the central site (CPU), in which case there is no need to solve a routing problem; all data between the terminal and CPU flows along this one path.

The two problems specified above are sometimes referred to as the concentrator location problem and the terminal layout problem, respectively, although there is no clear division between these. The concentrator location problem involves connection of n terminals T_i to one of m concentrators C_j or to the central site C_0 . Define:

C_{ij} - cost of the line connecting terminal i to concentrator j

d_j - cost of concentrator j and its high speed CPU line

$$x_{ij} = \begin{cases} 1 & , \text{ if } T_i \text{ is connected to } C_j \\ 0 & , \text{ otherwise} \end{cases} \quad (4-27)$$

$$y_j = \begin{cases} 1 & , \text{ if site } j \text{ contains a concentrator} \\ 0 & , \text{ otherwise} \end{cases} \quad (4-28)$$

The total cost

$$D = \sum_{i=1}^n \sum_{j=1}^m C_{ij} x_{ij} + \sum_{j=1}^m d_j y_j \quad (4-29)$$

Since each terminal is connected to one concentrator (or directly to the CPU)

$$\sum_{j=0}^m X_{ij} = 1 \quad (i=1,2,\dots, n) \quad (4-30)$$

Since the capacity of each concentrator is limited,

$$\sum_{i=1}^n X_{ij} \leq ky_j \quad (j=1,2,\dots, m) \quad (4-31)$$

If terminal connections to two or more concentrators are permitted, then (4-30) above can be modified in an obvious way, as can (4-31) if concentrators of different capacities are available.

The above formulation is similar to the plant, warehouse and facility location problems in operations research [B2] . Solutions normally employ either exhaustive search or branch-and-bound procedures [C3,S3,K3] . The problem involves integer linear programming requiring running times which tend to grow exponentially with the number of terminals. Extensions to layouts other than star (as we have assumed) are not easily accommodated. For these reasons, heuristic algorithms are usually employed for networks of practical size.

If concentrators are specified and located, then the term $\sum_{j=1}^m d_j y_j$ in (4-29) is known. In this case, it remains to assign terminals to concentrators. If each terminal is assigned to a unique concentrator, the problem is the transportation problem [B1,C3] for which methods of solution and performance

bounds are available.

The difficulties inherent in obtaining exact solutions for the concentrator location problem in practical situations motivate heuristic algorithms, which generate good solutions without excessive computation. The add [K4] and drop [F5] algorithms are two such algorithms.

The add algorithm begins with all terminals assigned to the central site C_0 at cost Z_0 . It then investigates each concentrator site individually, and selects for placement of the first concentrator that site for which solution of the terminal assignment problem yields the largest cost saving. The terminal assignment problem is easily solved if only one concentrator is available, and must be solved m times. All terminal assignments are then released, and the next concentrator is placed at one of the remaining $m-1$ sites. Again, the site which yields the largest (further) cost reduction is selected. The procedure continues until all concentrators have been located. Thus, if there are r concentrators and $m > r$ sites, the terminal assignment algorithm is solved approximately mr times. Investigation of all possible combinations of sites and concentrators requires m^r computations, a number which would normally be much larger than mr . The concentrator locations and terminal assignments resulting from application of the above procedure are further optimized by deleting some inefficiently used concentrators and reassigning their terminals, and by moving concentrators from existing to adjacent locations. Many variations are possible, in accordance with available computing power.

The drop [F5] algorithm proceeds in the reverse direction to the add algorithm. Initially, concentrators are located in all locations, and terminals are assigned to more than one concentrator. Concentrators and terminal-concentrator links are then dropped the order dictated by maximum cost reduction. Many variations in detailed execution of the drop algorithm are possible.

Both the add and drop algorithm, as well as the integer programming solution formulated earlier suffer from various drawbacks. First, multidropped lines are not easily handled. Second, constraints on time delay and reliability are not directly incorporated, although there is little scope for controlling reliability in a singly-connected centralized network. A delay constraint can be incorporated by limiting each concentrator's total input traffic to a fraction of that of the high-speed line between the concentrator and CPU in which case delay is obtained using the queueing equations in Chapter 3.

The add, drop and other centralized network design algorithms can be applied iteratively, with concentrators established during iteration k being regarded as terminals for iteration $k+1$. For the add and drop algorithms, the number of concentrator levels, as well as potential concentrator locations must be specified at the outset and do not therefore emerge as an inherent part of the design procedure. Although reliability is again virtually fixed at the outset, delay can be incorporated indirectly by limiting concentrator input-to-output traffic ratios. For example, if all messages are exponential with mean length μ , if nodal processing

times and propagation delays are negligible and if utilization factor ρ at each concentrator is identical, then the average delay T for terminal-to-CPU data is bounded by $N\rho/(1-\rho)$ where N is the number of levels of hierarchy. The fact that T may be less than $N\rho/(1-\rho)$ is due to the fact that not all terminals will be connected to the CPU via N levels of concentration; some terminals may be connected directly to the CPU. Similar reasoning can be used to bound the CPU-to-terminal delays.

IV-6 Topological Design of Centralized Networks Using Multidropped Lines

If all communication is between terminals and a single CPU, multidropped lines are often used instead of concentrators for reasons of lower cost. If the entire set of n terminals is to be connected via a single multidropped line, then Kruskal's [K5] algorithm can be used to minimize network cost. Execution of the algorithm, which yields a minimum spanning tree requires approximately n^2 operations. Suboptimum algorithms [K6] can be used to reduce the required number of operations. Prim's [P1] algorithm can also be used to obtain a minimum spanning tree using n^2 operations.

In many situations, time delay constraints will require that the number of nodes attached to any given line be bounded, in accordance with (3-49). Algorithms for generating minimum cost networks in these situations are available, but involve computation times which grow exponentially with the number of nodes [B1]. Typically, networks with more than 50 nodes require prohibitively large computation times.

Optimum algorithms facilitate determination of useful upper bounds on network cost using branch and bound techniques [B1, C4, E1, K7]. Lower bounds on network cost are obtained when constraints on the number of terminals per line are removed. Suboptimum algorithms are then used to design the actual network. A number of suboptimum algorithms are available, but network terminal arrangements can always be generated which confound the algorithms. The simplest heuristic involves a minor modification of Kruskal's algorithm. An additional modification by Esau and Williams [E2] improves the algorithm further. Virtually all existing heuristic algorithms are specializations of a general class of algorithms [K6] which order branches between nodes according to a biased cost, which may change as the algorithm proceeds. Determination of the bias defines the algorithm, which requires approximately $n \log n$ operations.

IV-7 Topological Design of Centralized Networks via Nodal Clustering

Clustering approaches to network design involve the following steps:

1. Grouping of nodes into clusters. Nodes in each cluster should be "similar" to each other in some sense, and "different" from nodes in other clusters.
2. Connection of the nodes in any one cluster to concentrators, multiplexors or multidrop lines.
3. Post-processing to further improve the network which results following steps 1 and 2 above.

Numerous clustering algorithms are available for a variety of applications [D3, K8, A1, K9]. We consider two algorithms which were used with subsequent design procedures to generate

networks with lower costs than those obtained using alternative techniques. One technique, called linear regression clustering (LRC) [D1] was developed and used to design singly-connected point-to-point networks consisting of terminals, concentrators and a CPU. A second technique was used to design singly-connected networks consisting of terminals on multidropped lines, concentrators and a central CPU [M1].

The LRC algorithm [D1] begins by employing linear regression to fit the collection of all nodes, denoted by cluster P_0 , by two lines, the X-line and the Y-line which is perpendicular to the X-line. The line perpendicular to the one which best fits the terminal locations weighted by the per-unit length costs of the data link of capacity sufficient to handle the terminal traffic defines the boundary between two new clusters P_1 and P_2 which are subclusters of P_0 . The new clusters are then further subdivided in the same way, until all clusters comprise terminals whose total traffic does not exceed that of the lowest capacity concentrator.

Each cluster is then optimized as follows. Cluster P_j is selected whose total terminal traffic does not exceed C_M , the capacity of the largest available concentrator, and which was obtained by subdividing a cluster whose total traffic exceeds C_M . The cost of the terminal-concentrator links, concentrator-CPU link and concentrator itself is then totalled, following selection of that terminal location at which the concentrator should be located to minimize this total cost. The same calculation is then repeated for all subclusters of P_j , using concentrators whose capacities are the minimum sufficient to handle the

cluster's total terminal traffic. The final design for cluster P_j consists of that combination of concentrators and terminals which minimizes total cost for cluster P_j . Thus, in one situation a single concentrator of capacity C_M may be assigned to cluster P_j . In another situation, P_j may consist of several smaller concentrators assigned to the various subclusters of P_j .

The final step in design involves further cost reductions by:

1. Moving all concentrators from terminal locations to cluster medians.
2. Replacing concentrator-terminal connections by terminal-CPU connections whenever further cost savings result.

The LRC algorithm was evaluated by computer design of networks having from 100 to 500 terminals. The costs and performance capabilities of the resulting networks were compared with networks designed using the add algorithm which is usually employed for centralized network design problems. In each case the same randomly generated sets of terminal locations were used. In comparison with the add algorithm, the LRC algorithm showed the following advantages:

1. The cost of the LRC networks was 8 percent less.
2. The average delay was 40 percent less.
3. The cost of adding new terminals and data links is typically 50 percent less.
4. The computational cost of design is typically 20 times less for 100 node networks, 85 times less for 300 node networks and 150 times less for 500 node networks.

Examination of the resulting networks showed that the LRC networks contain up to 50 percent more concentrators than

networks designed using the add algorithm. This larger number of concentrators reduces delay and facilitates the addition of new terminals to the network without large cost increases which otherwise result from saturation of existing concentrator capacity. As concentrator costs decrease relative to data link costs, the relative attractiveness of the LRC algorithm increases.

The LRC study [D1] demonstrates that the number, location and interconnection of concentrators should not be specified at the outset, but should evolve as an inherent part of the design procedure.

A difficulty in the LRC algorithm appears following the happy realization that the algorithm can be reapplied after the first level of concentrators have been specified, located and assigned terminals. These concentrators can then be regarded as nodes with traffic equalling the associated terminal traffic, and the LRC algorithm can be reapplied to yield second and subsequently higher levels of concentration. The difficulty involves control of delay and, to a lesser extent, reliability during design. As explained in Section 4-6 and 3-3, delay depends on the number of levels of concentration and link utilization, neither of which would be specified at the outset. One approach would be to guess at the number of levels of hierarchy, and require that equal delay at each level. The concentrator utilization could then be specified to meet delay requirements. Design could be repeated for various initial guesses as to the eventual number of hierarchical levels. There are undoubtedly better approaches which are awaiting discovery.

A different clustering algorithm was developed and used by

MacGregor and Shen [M1] to design multidrop point-to-point networks. Clusters of terminal nodes are formed by "rolling balanced snowballs". Thus, two nodes closest together are selected. If these two can be put in the same cluster without overloading the available multidrop line capacity, they are temporarily replaced by a single node at their centre-of-mass. This newly formed COM node has weight equal to the number of nodes in the cluster. The clustering process continues until no remaining pairs of nodes, most of which will be COM nodes, can be merged.

Following the clustering procedure, the add algorithm is employed to assess each COM node as a GAF (concentrator) site. With those sites showing the greatest cost benefits are associated the neighbouring COM nodes. The GAF sites are then moved to the best (in terms of overall cost) terminal locations, and a multidrop line layout algorithm is used to link the terminals in each cluster to the GAF which is connected directly to the CPU.

The performance of the algorithm was evaluated by designing networks and comparing the resulting network costs with those of networks designed using the best previously developed algorithm, the average tree-direct (ATD) algorithm [W1]. The ATD algorithm converts the multidrop line layout problem to a point-to-point problem by forming a cost matrix based on the average of the cost of connecting two nodes through a minimum spanning tree and the cost of connecting the nodes directly to the CPU [W1]. Both the COM and ATD algorithm used the add algorithm for GAF site location, both used the same multidrop line layout procedure and both were programmed using similar programming techniques [M1].

Specified at the outset were node locations and traffic,

as well as the number of nodes per line, number of lines per GAF, line and termination costs, and GAF costs. Randomly distributed sets of 50, 100, 200, and 400 nodes were considered, as were nodes randomly distributed on the basis of population in the USA.

The ATD networks showed a cost approximately 5 percent higher than the COM networks, except when limits to the number of lines per GAF facility was removed, in which case the cost difference was 0.5 percent. Network cost increased linearly with the number of nodes when GAF's of fixed capacity were used. The COM algorithm showed a lower program execution cost, particularly for networks having a large number of nodes. The COM and ATD execution costs (EC) were approximated by the following equations, where N is the number of network nodes.

$$EC = (2 \times 10^{-4}) N^{-2} \quad (\text{COM algorithm}) \quad (4-32)$$

$$EC = 10^{-4} N^{-2.5} \quad (\text{ATD algorithm}) \quad (4-33)$$

The work [M1] referred to above did not directly consider delay or reliability, although average delay seen by the terminals could be calculated from the multidrop line delay equation (3-49). To this delay is added any delay resulting from queueing of messages from different multidrop lines at the concentrators (GAF's).

Calculation of multidrop line reliability is straightforward in principle; failure probability p_i in equation (2-13) must be modified in recognition that terminal i is active only when it is transmitting or being polled. The centralized nature of multidropped networks implies that little flexibility is availability to design for reliability constraints.

The linear cost-vs-number of nodes behaviour in [M1] may seem to contradict anticipated economies of scale. Such economies would undoubtedly occur if several GAF levels were used or if available GAF capacity increased with the number of terminals.

The COM algorithm ignores node traffic levels in forming clusters. Refinement to accommodate large traffic level variations could lead to better networks when such variations are present.

Direct comparisons of network cost and performance of multidropped networks against networks using single-dropped links and concentrators seem unavailable.

IV-8 Topological Design of Distributed Networks: Non-Clustering Approaches

In a distributed network, most or all nodes are connected to many other nodes, and several alternate paths are available for traffic flowing between two nodes.

Several techniques, all of them heuristic are available for the topological design of distributed networks. Design involves specification of topology, link flows $\{\lambda_i\}$ and link capacities $\{C_i\}$. Constraints include average delay T , flow conservation at each node, $\lambda_i \leq \mu C_i$, and usually a heuristic reliability constraint; for example, that the network be two-connected.

The branch exchange method (BXC) has been employed to design natural gas pipelines [G2], as well as survivable networks [S4] and computer networks [F1]. The method starts with an arbitrary topological configuration, adds a new link and removes an old link to maintain two-connectivity. Capacities and flows are then optimized. If a cost-throughput improvement results, then this topological transformation is retained. Otherwise it is rejected.

If all local transformations have been explored, the topological configuration is not modified further. Either a new starting topology is examined, or the best of the existing networks is retained as the solution to the optimization problem.

The concave branch elimination method (CBE) [G2, Y1] is applicable whenever actual link cost-vs-capacity curves are approximated by concave functions. A fully connected network is selected at the outset, with flows $\{\lambda_i\}$ and capacities $\{C_i\}$ chosen to meet all constraints. The flow deviation (FD) algorithm is then applied until a locally optimum network results. Once the FD algorithm has assigned a branch zero flow, the flow and therefore the capacity remains at zero during all subsequent flow iterations. Consequently, many branches are eliminated during execution. If K-connectivity is required, the algorithm is terminated whenever the next iteration violates the K-connectivity constraint. To obtain several locally optimum networks, several starting flows are used. Once locally optimum networks are obtained further optimization can be used to specify actual (discrete) link capacities and correspondingly modified flows.

The cut-saturation (CS) algorithm [G3, B1] can be regarded as a modification of the BXC algorithm whereby only those branch exchanges which are likely to result in better networks are considered. The selection of potentially beneficial modifications involves successive removal of links for which $\lambda_i \ll \mu C_i$, followed by rerouting of data over remaining links. Saturated cuts (a cut is an isolated set of links whose removal disconnects the network; a cut is saturated if $\lambda_i \simeq \mu C_i$ for each link in the cut) are then identical, and links are then added to these cuts. The fact that

most of the possible branch exchanges are removed from consideration permits near-optimal routing to be implemented at each stage of the CS algorithm.

The CS algorithm yields networks which are usually better than those obtained from the BXC algorithm and comparable to those obtained using the CBE algorithm [B1, G2]. The computational time for the CS algorithm is much better than that for the BXC algorithm and comparable to that for the CBE algorithm.

Lower bounds on cost for a given network throughput can be obtained by approximating link cost-vs-capacity curves by lower envelopes. If links having these approximate characteristics are then used in a fully connected network, then the desired lower bounds on network cost are obtained by joint optimization of the link flows and capacities, without regard for connectivity constraints.

Application of heuristic design methods to the 26-node ARPANET topology yields interesting results. First, the three different methods cited above yield networks with different topologies, but similar costs. Second, different methods yield the best networks, depending on the throughput γ . Finally, the cost of heuristic methods tends to fall within 15% of that for networks obtained using bounds as described in the previous paragraph.

For an Arpanet nodal topology with $\gamma = 650$ kbs, the CS method seems best and yields a network having approximately 30 links and a cost of \$0.135 bits/sec./month. With $\gamma = 700$ kbs the CBE method seems best and yields a network with approximately 60 links and a cost of \$0.127 bits/sec./month. A natural conclusion is that there are many good solutions to a network design problem, and that

the good solutions may differ considerably in topology, capacity assignment and routing.

Virtually no information is available concerning reliability comparisons, or the effects on the design of a formal reliability constraint.

The methods cited above do not generate new nodes not specified at the outset of the design. It is conceivable that generation of these additional nodes during design could further reduce the overall network cost, in the same way that generation of additional nodes for concentrator location reduces the cost of centralized networks.

IV-9 Location of Backbone Nodes

Backbone nodes interface local access networks to a distributed high-speed data network [H3, R1]. Thus, each backbone node is part of the distributed network and is also the central node of a local access network. Attention to the backbone node location problem is relatively recent.

When potential backbone node locations are specified, the add algorithm approach can be used to locate these to minimize the sum of local access costs, distributed network costs and backbone switch costs [H3]. At each iteration the potential switch sites are successively evaluated, and the one yielding the largest cost reduction is selected. This procedure is then repeated by selecting the next best site from those remaining. In applying the above algorithm Hsieh et al [H3] found the network cost to be rather insensitive to the number of switches used in the neighbourhood of the optimum; little change in cost occurred as this number varied

from 5 to 12 in a typical network design problem.

Although the drop algorithm [F5] approach seems not to have been used to locate backbone nodes, it would seem to be a viable alternative approach. Initially, all potential locations would be assigned backbone nodes, to which would be centrally connected local nodes. Backbone nodes would then be deleted one at a time, with the corresponding local nodes being re-assigned. Those nodes offering the greatest overall network cost reduction would be selected for deletion.

Clustering of nodes has been used for backbone node location [H3]. If one assumes that local access costs dominate total network costs, then backbone switches should be located near the centre of mass of the clusters' nodes to minimize local access costs. The remaining problem is to define the clusters. An approach similar to the one used by MacGregor and Shen [M1] was used by Hsieh et al [H3] to select the best N backbone node locations for various N, and the total cost of networks subsequently designed was found to be approximately 5% higher than networks designed using the add algorithm approach, and relatively insensitive to variations in N about the optimum value.

Networks optimized following either random or manual selection of backbone sites were found to have substantially higher costs than those obtained using both add and cluster design approaches. Thus, the relative insensitivity to the number of backbone switches does not imply that their actual location is unimportant.

Computation design costs favour the add approach for smaller networks and the cluster approach for larger networks.

The backbone node location problem results from the partitioning of the network design problem into local access design and distributed network design. Such a partitioning replaces a complex design problem by several simpler problems, but normally eliminates many design options. Thus, the heirarchical structure of the network is constrained as a first step in design as backbone nodes are restricted as to number and location.

In many situations, traffic matrixes detailing traffic flow from each node in one local region to each node in all other regions would be unavoidable and, if available, unwieldy. Traffic patterns from area-to-area, and within each area would be used as a basis for design. One might therefore argue that partitioning the network design problem into local access design and distributed network design will avoid crippling non-optimalities. The counter-argument is that such a procedure severely constrains the network's heirarchical structure at the outset, and that this structure should evolve as an inherent part of the design procedure which would be based on both area-wide and local traffic patterns.

IV-10 Topological Design of Distributed Networks Using Nodal Clustering

The LRC algorithm developed for the design of centralized networks was recently extended to the design of singly-connected distributed networks [D4]. Nodes of specified location and traffic r_{ij} were clustered as explained in Section 4-7. Clusters were then optimized by connecting nodes in subclusters to

concentrators, which were then connected to concentrators in neighbouring subclusters to minimize total cost of concentrators and data links. Concentrators so established were then regarded as new nodes with traffic equal to that of the nodes connected to the concentrator, and the optimization was repeated until a singly-connected network resulted. The resulting network showed cost-vs-throughput characteristics 20% better than those resulting from a non-cluster design approach based on a search for good networks [D4]. The execution time of the cluster design algorithm was considerably better than that of the non-clustering approach.

The clustering algorithm referred to above needs further refinement. Direct control of average delay is presently difficult, since the number of levels of network hierarchy, on which delay depends is itself known only at the completion of the network design. The algorithm has been implemented for singly-connected networks only; extension to multiconnected networks seems possible at the expense of increased computation cost. Reliability was not incorporated as a design constraint, although reliability as given by (2-13) was calculated following completion of the network design, and compared favourably with that of networks designed using the non-clustering approach. Incorporation of (2-13) as a constraint would seem to present no significant difficulties.

As noted in Section 4-3, routing tables at each node increase linearly with the number of nodes N and with the connectivity of the network. Unless some type of hierarchical routing scheme is used, the size of the routing tables will result in

prohibitive nodal buffer storage costs and processing delays, as well as possibly prohibitive traffic overheads arising from routing table updates. The problem is illustrated in Fig. 4-1, where node 1 in each of four areas provides for access to nodes in three other areas. The destination address of any message would be represented by two two-bit numbers, the first of which would specify the area and the second the local node. Thus, routing tables at local nodes 2, 3, or 4 would need only three entries per eligible outgoing link, while tables for any node 1 would need six entries per eligible outgoing link. If all traffic for a given destination were restricted to a single outgoing link (which could be changed adaptively), a total of $3 \times 4 + 6 \times 4 = 36$ routing entries would be needed for the entire network. If non-heirarchical routing were used, $15 \times 16 = 240$ entries would be required.

Heirarchical routing has been considered recently by Kamoun [K8, K10] who showed for a class of "balanced" networks that the minimum routing table length is achieved when $m = \ln N$ heirarchical clusters are used, in which case the average routing table length $l \approx \ln N$. The resulting increase in communication path length as measured by the average increase in the number of links traversed in comparison with non-heirarchical routing was obtained for symmetrical torus networks of varying connectivity. Even for non-optimum values of $m=2$ or $m=3$, the size of the routing table decreases considerably with only moderate path length increases. The higher the connectivity, the less the path length increase; here one sees a reason for high connectivity which seems to have not been heretofore appreciated. Typically, for $N=10^4$ and $m=3$, a 10^{-2} reduction in table length results at the expense of a path

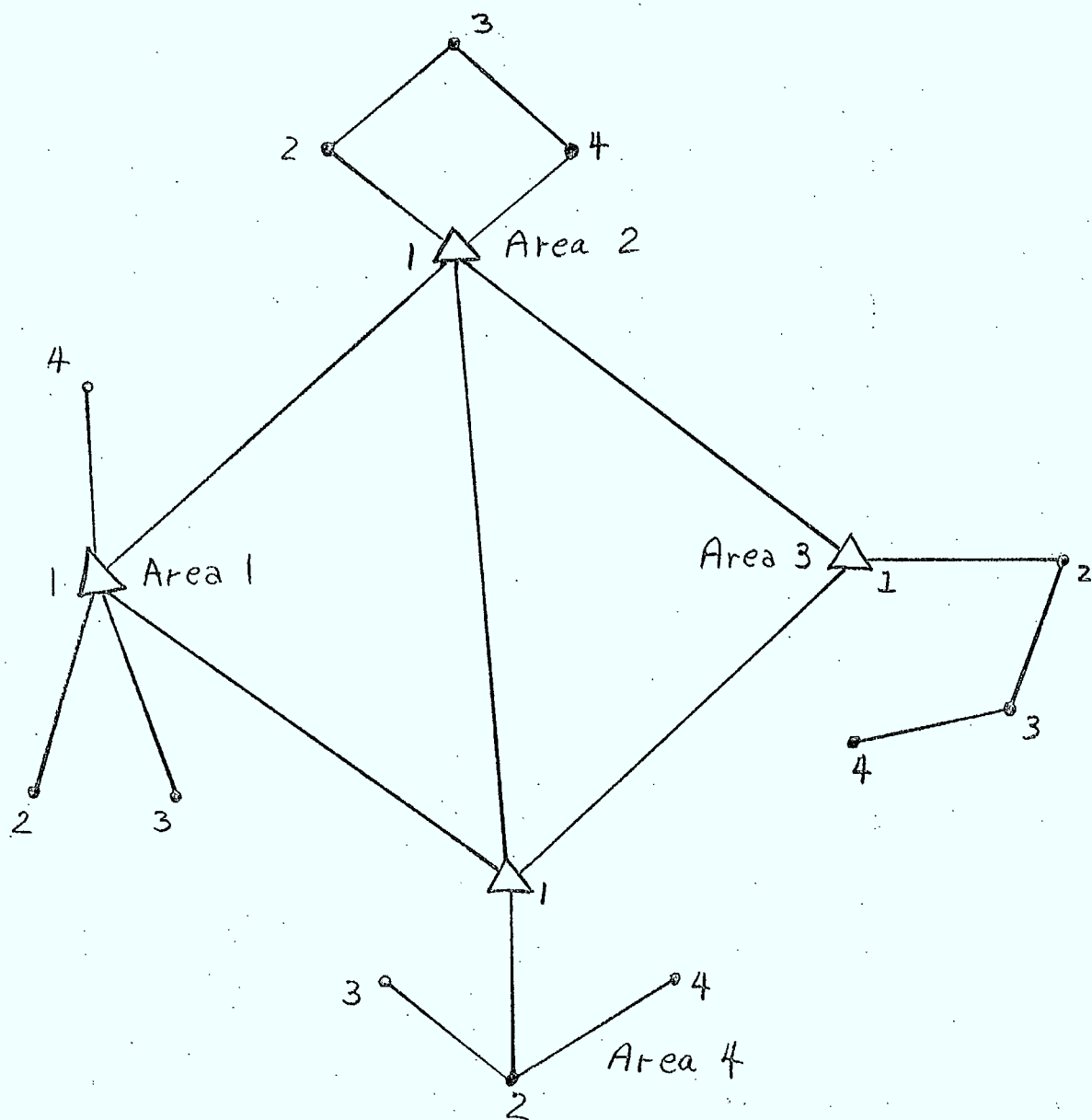


Fig. 4-1 Illustrating hierarchical routing.

length increase of approximately 50% [K8].

Kamoun [K8] also calculated degradations in delay and throughput resulting from path length increases, assuming symmetrical networks with equal flows and capacities in all links, and equal nodal delays. Effects of routing update traffic was included by using various relationships between update traffic and network size. Effects of nodal capacity constraints were also considered.

Although delay and throughput degradations did accompany heirarchical routing, these were moderate, did not normally exceed factors of two, were largest at $N \approx 200$ and approached zero for large N . The actual degradations are dependent on the degree of connectivity, number of cluster levels, and assumptions regarding update traffic level vs network size.

Although Kamoun's quantitative results are based on unrealistically idealized network symmetrics and traffic levels, his work clearly demonstrates that distributed networks must use heirarchical routing. It remains to develop network design procedures which incorporate routing table storage, associated processing delays and routing update traffic overhead as part of the design procedure. Designs which employ nodal clustering as a first step seem appropriate, although the basis on which clusters are to be defined is not altogether clear at this point. Heirarchical routing considerations would seem to favour clusters consisting of maximum intra-cluster traffic and minimal inter-cluster traffic [K8], where as non-routing considerations would seem to favour clustering on the basis of geographical proximity and traffic level [D1, D4].

V-11 Effects of Traffic Level on Network Costs

Of considerable interest and importance is the dependence of network cost on traffic throughput γ .

Assume for the moment that all entries r_{ij} in the traffic matrix $[r]$ are increased by the factor K , and that flows $\{\lambda_i\}$ and link capacities $\{C_i\}$ are also increased by K . Queueing delays at any node actually decrease by factor K as explained in Section 3-3, although nodal processing delays may increase somewhat because of the increased throughput. Unreliability U as defined by (2-13) will increase in proportion to increases in p_i which may accompany increases in capacity, as explained in Section 2-4. Since link and node costs will usually increase more slowly than capacity, the cost per bit transmitted will decrease as overall throughput γ increases. For this reason and because capacities need not increase as fast as γ to meet existing constraints, networks would be expected to exhibit positive economies of scale, in so far as capital costs are concerned. We have yet to deal with management of existing networks, which involve operating costs, including overhead traffic.

Confirmation regarding the above costs-vs-throughput behaviour is available. For example, Diriltzen and Donaldson [D4] show a total communication cost per kilobit of traffic of 150 for 100 node networks, 110 for 300 node networks and 80 for 500 node networks whose nodes are randomly positioned on a unit square with equal and constant average traffic per node. These costs do not change very much when both the average traffic per node and area covered increase linearly with the number of nodes [D4]. Gerla and Kleinrock [G2] report \$0.15/bits/sec./month for a 480 Kbs

throughput.

Quantitative results showing dependence of reliability on throughput seem unavailable.

IV-12 Unresolved Problems in Network Design

The following are among the unresolved network design issues:

1. How does one determine the optimum topology, data link capacity and link flows, given the external node locations and corresponding traffic matrix?
2. What are the performance characteristics of the optimum network?
3. How should heirarchical routing considerations be formally incorporated into the network design procedure?
4. What is the effect of formally incorporating a reliability constraint such as (2-13) into existing as well as new design procedures?
5. How should networks be designed to easily accommodate new users or changing traffic characteristics of existing users.

Regarding items 1 and 2 and centralized networks, the only way to know whether to use concentrators or multidropped lines is to actually design both types of networks to meet stated constraints, and to then compare the performance of the designs. It may be desirable to use both types of access in a single network; however, studies detailing possible design approaches and resulting network performance seem unavailable.

Items 1-3 above relate to distributed network design. Existing evidence indicates that a network's heirarchical structure should not be specified a-priori, but should emerge from execution of the design algorithm, which should include buffer storage costs

and delays which result from the nodal storage of routing tables. Design of efficient heirarchical systems is a difficult problem which arises in other contexts including multivariable control, transportation networks and governmental structures.

Heirarchical routing favours networks having high connectivity, in order to avoid large path length increases and accompanying degredations in delay and throughput, as explained in Section 4-10. Cost considerations seem to favour some sort of compromise between high connectivity, which implies short links and low link capacity, and low connectivity which favours a few links which, however, have high capacity. Well-designed low and high connectivity networks seem to yield similar costs, throughputs and delays.

Reliability as defined by (2-13) also indicates a compromise between high and low connectivity networks. High connectivity implies low values of λ_i and p_i due to small flows on many short links; low connectivity implies the opposite effects. Where the compromise lies is unclear, particularly if nodal reliabilities are included.

Design for accommodation to changing network usage seems essential, but seems to receive little more than cursory consideration. One approach is to consider the effects that changes in design data would have on the performance of an existing network. A more desirable approach would include adaptability as an initial design constraint. Adaptability is important in order that networks currently under development or in operation can take advantage of new developments which will undoubtedly occur during the coming decades.

IV-12 References

- AI M.R. Anderberg, Cluster Analysis for Applications. New York, N.Y.: Academic Press, 1973.
- B1 R.R. Boorstyn and H. Frank, "Large-scale network topological optimization," IEEE Trans. Commun., vol. COM-25, pp. 29-48, Jan. 1977.
- B2 L.R. Bahl and D.T. Tang, "Optimization of concentrator locations in teleprocessing networks," in Proc. Symp. on Computer Communications and Teletraffic, J. Fox, Ed., Brooklyn, N.Y.: Polytechnic Institute of Brooklyn Press, April 1972, pp. 355-362.
- C1 D.G. Cantor and M. Gerla, "Optimal routing in a packet-switched computer network," IEEE Trans. Comput., vol. C-23, pp. 1062-1069, Oct. 1974.
- C2 W.W. Chu, Ed., Advances in Computer Communications. Dedham, Mass.: Artech House, 1976.
- C3 L. Cooper, "The transportation location problem," Oper. Res., vol. 20, pp. 94-108, Jan.-Feb. 1972.
- C4 K.M. Chandy and R.A. Russell, "The design of multipoint linkages in a teleprocessing tree network," IEEE Trans. Comput., vol. C-21, pp. 1062-1066, Apr. 1972.
- D1 C.B. Danzig, Linear Programming and Extensions. Princeton, N.J.: Princeton Univ. Press, 1963.
- D2 H. Diriltten and R.W. Donaldson, "Topological design of teleprocessing networks using linear regression clustering," IEEE Trans. Commun., vol. COM-24, pp. 1152-1159, Oct. 1976.
- D3 R.O. Duda and P.E. Hart, Pattern Recognition and Scene Analysis. New York, N.Y.: Wiley, 1973.
- D4 H. Diriltten and R.W. Donaldson, "Topological design of distributed data communication networks using linear regression clustering," IEEE Trans. Commun., Oct. 1977.
- E1 D. Elias and M.J. Ferguson, "Topological design of multipoint teleprocessing networks," IEEE Trans. Commun., vol. COM-22, pp. 1753-1762, Nov. 1974.
- E2 L.R. Esau and K.C. Williams, "On teleprocessing system design, Part II," IBM Syst. J., vol. 5, pp. 142-147, 1966.
- F1 H. Frank, I.T. Frisch, W. Chou, and R. Van Slyke, "Optimal design of centralized computer networks," in Networks, Vol. 1. New York: N.Y., Wiley, 1971, pp. 43-57.

- F2 H. Frank and W. Chou, "Topological optimization of computer networks," Proc. IEEE, vol. 60, pp. 1385-1397, Nov. 1972, also in [C1].
- F3 H. Frank and I.T. Frisch, Communication, Transmission and Transportation Networks. Reading, Mass.: Addison-Wesley, 1971.
- F4 L. Fratta, M. Gerla, and L. Kleinrock, "The flow deviation method: an approach to store-and-forward communication network design," Networks, vol. 3, pp. 97-133, 1973.
- F5 E. Feldman, F.A. Lehner and T.L. Ray, "Warehouse location under continuous economies of scale," Management Science, vol. 12, pp. 670-684, May 1966.
- G1 M. Gerla, "The Design of Store and Forward Networks for Computer Communications," School of Enggr. and Appl. Sc., UCLA Ph.D. Thesis, University of California, Los Angeles, Jan. 1973.
- G2 M. Gerla and L. Kleinrock, "On the topological design of distributed networks," IEEE Trans. Commun., vol. COM-25, pp. 48-61, Jan. 1977.
- G3 M. Gerla, "A cut saturation algorithm for topological design of packet switched communication networks," in Proc. Nat. Telecommun. Conf., Dec. 1974, pp. 1074-1085.
- H1 D. Huynh, H. Kobayashi and F.F. Kuo, "Optimal design of mixed-media packet-switching networks: routing and capacity assignment," IEEE Trans. Commun., vol. COM-25, pp. 158-169, Jan. 1977.
- H2 T.C. Hu, Integer Programming and Network Flows. Reading, Mass.: Addison-Wesley, 1969.
- H3 W. Hsieh, M. Gerla, P. McGregor and J. Eicki, "Locating backbone switches in a large packet network," in Conf. Rec., Third Inter. Conf. Comput. Commun., Toronto, Canada, Aug. 1976, pp. 330-339.
- K1 L. Kleinrock, Queueing Systems, Volume 2: Computer Applications. New York, N.Y.: Wiley, 1976.
- K2 L. Kleinrock, Communication Nets: Stockastic Message Flow and Delay. New York, N.Y.: McGraw-Hill, 1964.
- K3 B.M. Khumawala, "Warehouse location problems, efficient branch and bound algorithm," Management Sci., vol. 18, pp. B718-B731, Aug. 1972.
- K4 A.A. Kuehn and M.J. Hamburger, "A heuristic program for locating warehouses," Management Science, vol. 9, pp. 643-666, July 1963.

- K5 J.B. Kruskal, "On the shortest spanning subtree of a graph and the travelling salesman problem," Proc. Amer. Math. Soc., vol. 7, 1956.
- K6 A. Kershenbaum and W. Chou, "A unified algorithm for designing multidrop teleprocessing networks," IEEE Trans. Commun., vol. COM-22, pp. 1762-1772, Nov. 1974.
- K7 A. Kershenbaum and R.R. Boorstyn, "Centralized teleprocessing network design," in Conf. Rec., Nat. Telecommun. Conf., New Orleans, Miss., Dec. 1975, pp. (27-11)-(27-14).
- K8 F. Kamoun, "Design Considerations for Large Computer Communication Networks," UCLA Comp. Science Dept. Rept., UCLA-ENG-7642, University of California, Los Angeles, California, 1976.
- K9 B.W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," Bell Syst. Tech. J., vol. 49, pp. 291-307, Feb. 1970.
- K10 F. Kleinrock and F. Kamoun, "Hierarchical routing for large networks, performance evaluation and optimization," Computer Networks, vol. 1, pp. 155-174, 1977.
- M1 P.V. McGregor and D. Shen, "Network design: an algorithm for the access facility location problem," IEEE Trans. Commun., vol. COM-25, pp. 61-73, Jan. 1977.
- P1 R.C. Prim, "Shortest connection networks and some generalizations," Bell Syst. Tech. J., vol. 36, pp. 1389-1401, 1957.
- R1 R.D. Rosner, "Large scale network design considerations," in Conf. Rec., Third Inter. Conf. Comput. Commun., Stockholm, Sweden, Aug. 1974, pp. 189-197.
- S1 M. Schwartz, Computer-Communication Network Design and Analysis. New York, N.Y.: Prentice-Hall, 1977.
- S2 M. Schwartz and C.K. Cheung, "The gradient projection algorithm for multiple routing in message-switched networks," IEEE Trans. Commun., vol. COM-24, pp. 449-456, Apr. 1976.
- S3 K. Spielberg, "Algorithm for the simple plant location problem with some side conditions," Oper. Res., vol. 17, pp. 85-111, Jan. 1969.
- S4 K. Steiglitz, P. Weiner and D.J. Kleitman, "The design of minimum cost survivable networks," IEEE Trans. Circuit Theory, vol. CT-16, pp. 455-460, Nov. 1969.
- W1 L.S. Woo and D.T. Tang, "Optimization of teleprocessing networks with concentrators," in Conf. Rec., Nat. Telecommun. Conf., Atlanta, Ga., Nov. 1973, pp. 37C1-37C5.
- Y1 B. Yaged, Jr., "Minimum cost routing for static network models," Networks, vol. 1, pp. 139-179, 1971.

V OPERATION AND MANAGEMENT OF NETWORKS; NETWORK PROTOCOLS

V-1 Network Management Issues

Network operation and management as well as design of reliable and efficient network protocols is much less well understood than is network design as considered in Chapter 4. In the present chapter, some of the important management and protocol issues are discussed, and relationships between the issues are indicated, insofar as these are presently understood.

Most of the discussions pertain to the delivery of individual message packets from source to destination. Sections 4-2 and 4-3 deal with the reliable node-to-node delivery of individual packets. Section 4-7 considers the source-destination delivery of packets.

Sections 4-4, 4-5 and 4-6 discuss priority queuing, adaptive routing and flow control. The subject matter is inter-related and involves the smoothing of traffic irregularities and provision of (hopefully) satisfactory service to network users.

Section 4-8 considers special problems for broadcast radio networks, which are of growing interest.

The writing of this chapter has been both exciting and exasperating. New insights which continually present themselves while reading the literature and extracting the important issues, ideas, and advances provide the excitement. The exasperation results from the realization that more time spent on study would yield additional insights and understanding of a vast subject matter whose importance grows with the growth in EIS networks, applications and users. As noted in Section 1-3, a large fraction

(approximately 90 percent in a lightly loaded ARPANET) of network traffic is management overhead [K1, K2, K3].

V-2 Data Link Controls

Transmission of a sequence of data symbols from one node to another cannot begin or conclude without some sort of control procedure. The procedure must include the means for a sender to either request permission for or advise of transmission, and to conclude transmission. The actual transmission must be synchronized at the word (frame) level, and errors must be detected and corrected. It is generally agreed that the sole task of a data link control (DLC) is to provide for node-to-node delivery of error-free messages or packets [G1, S1, K1]. DLC's provide for conversion of a transmission link into a communication link.

Several approaches are available for implementing word synchronization. One involves use of a prefix consisting of a sequence of symbols, which sequence does not naturally occur in any sequence consisting of prefix followed by text followed by prefix. The disadvantage of such a scheme is that the prefix required is of length approximately equal to that of the text [S2]. An alternative is to use a prefix with favourable correlation properties; Barker sequences are often used for such purposes [S2, M1]. A received symbol stream is correlated against the prefix, and the location of correlation peaks establishes synchronization. Successful application of the method requires that successive data blocks be independent. Considerable synchronization delay may result, but a short prefix can be used. The method is not suitable when a single packet is to be transmitted, as in an interactive conversation.

Perhaps the most common word synchronization procedure involves use of a prefix whose otherwise occurrence is prevented by bit stuffing [S1, G2]. For example, the prefix 0111110 may be selected, and its uniqueness is guaranteed by following any sequence of four consecutive one's in the text by a zero inserted at the transmitter and removed by the receiver.

Consider a sequence of binary symbols consisting of the 011...10 prefix having m ones, followed by the k text bits. The maximum number of artificially stuffed zeros is $\lceil k/(m-1) \rceil$ where $\lceil X \rceil$ denotes the smallest integer which exceeds X . The ratio $g(m,k)$ of the maximum number of synchronization bits to the total number of bits, including text bits plus synchronization bits is

$$g(m,k) = \frac{(m+2) + \lceil k/(m-1) \rceil}{(m+2) + \lceil k/(m-1) \rceil + k} \quad (5-1)$$

Differentiation of (5-1) with respect to m shows that $m = 1 + \sqrt{k}$ minimizes g , and that the resulting minimum value is

$$g(1 + \sqrt{k}, k) = \frac{\sqrt{k} (\sqrt{k} + 3) + k}{\sqrt{k} (\sqrt{k} + 3) + k (\sqrt{k} + 1)} \quad (5-2)$$

For large k , g in (5-2) approaches $1/\sqrt{k}$; as k becomes infinite, g approaches zero.

For typical finite values of k , g can be made reasonably small for appropriately selected values of m . For example, if $k = 100$, the optimum value for m is 11, in which case $g(11, 100) = 0.187$. With $k=900$, the optimum $m=31$, in which case $g(31, 900) = 0.065$. Thus, a 9-fold increase in text length results in a 3-fold

decrease in synchronization overhead. Here we see confirmation of the reduced overhead which accompanies an increase in text or packet length, as noted in Section 2-5.

We have tacitly assumed that only one prefix is used per text segment, and that a text segment plus prefix constitutes a packet. If a prefix also terminates a packet then an analysis similar to that above shows that $h(m,k)$, the ratio of the maximum number of synchronization symbols to synch plus text symbols is

$$h(m,k) = \frac{2(m+2) + \lceil k/(m-1) \rceil}{2(m+2) + \lceil k/(m-1) \rceil + k} \quad (5-3)$$

In this case $m = 1 + \sqrt{k/2}$ minimizes h which approaches zero as k becomes infinite. For $k=100$ and 900 , respectively, the optimum values for m are 8 and 22. The resulting values of h are 0.259 and 0.092, respectively.

The above analysis deals with minimization of the synch overhead when such overhead is maximized due to bit stuffing following every $m-1$ text bits. The average overhead would, of course, be less, with the result that shorter prefixes than those indicated above would minimize average overhead. The actual calculation of the average overhead seems difficult, and may require simulation techniques.

The above discussion also implies an absence of symbol transmission errors. Errors may obliterate synchronization prefixes and may cause the spurious occurrence of prefixes in the text portion of the transmitted sequence [S2]. Use of cyclic redundancy checks [L1, L2] provides for the detection of such errors. Some consideration has been given to designing codes with parity check bit

constraints which enables both error detection (and/or correction) as well as synchronization, however the efficiency (in terms of overhead) of these codes is rather low [S2, M2, S3, H1, S4, T1].

Different systems use different frame formats for data link control [S1, K1, K2]. Recent efforts at standardization has resulted in the format shown in Fig. 5-1 [S1]. The American National Standards Institute's (ANSI), Advanced Data Communication Control Procedure (ADCCP) as well as the International Standards Organization's (ISO) High-Level Data Link Control (HDLC) and IBM's Synchronous Data Link Control (SDLC) all utilize the format in Fig. 5-1. Frame synchronization is established with the aid of bit stuffing, as explained above. The address and control fields can be expanded in multiples of eight bits. The 16-bit block check uses the CCITT standardized polynomial $g(X) = X^{16} + X^{12} + X^5 + 1$ and checks the address, control and information bit sequence.

Three types of frames are used; information transfer (I) frames, supervisory (S) frames, and unnumbered control (U) frames. The first is used for transmission of user information to the specified address. The control field is used to record unacknowledged packets. The S and U type frames are used for data link control tasks as described at the beginning of this section.

In addition to word synchronization, which has been discussed, and error control which is considered in the next section, a DLC must advise the receiving node of an impending message, and advise regarding conclusion of transmission. The variety of ways in which these functions are implemented are seen upon examination of the various existing and proposed DLC's [K1, R1, S1]. For example, a DLC can either request permission to transmit a message or packet,

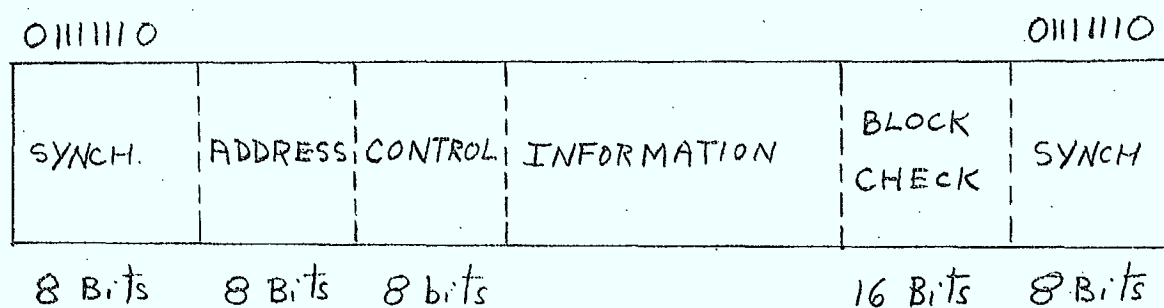


Fig. 5-1 Standardized word (frame) format for data link control.

and transmit upon receipt of such permission. Alternatively, the DLC may transmit the message itself, in which case the receiving node is merely advised of transmission. Messages which cannot be accommodated, possibly because of filled nodal buffers, must be retransmitted. The second alternative seems most efficient unless many retransmissions are required.

There is no established procedure for analysis of DLC's, which should be both reliable and efficient [K1, K2, S5, S6]. Reliability implies that with high probability messages move from node to node as desired; efficiency implies that a small fraction of the actual traffic is for control purposes, and that message delay and throughput are not impaired. Reliability analysis normally involves either an exhaustive listing of all possibilities [B1, S6, R1], or the use of a state diagram (finite state machine) representation [D1, M3, M4, S6, G2, G3] to more readily determine the various possibilities, including those which involve improper link operation.

Efficiency analysis involves quantitative determination of how network cost as well as delay, reliability and throughput for actual message information are affected by a DLC. Once the DLC has been articulated and analysed for reliability, its quantitative analysis for efficiency seems relatively easy, as indicated above for synchronization and in the next section for error control. However exact closed form solutions may have to be replaced by approximations.

An alternative approach to DLC evaluation involves either performance measurements during actual network operation [K1, K2, K3] or measurements on simulated networks. While this latter

approach obviates the necessity of actual analysis, and would eventually be performed in any case, its early use has several disadvantages. These include measurement cost, lack of insight, and measurement errors resulting from an insufficient number of observations of some atypical but troublesome situations.

Analogies exist between data link controls and communication between people. Although constrained natural languages have been widely studied [G2, G4, H2], the useful application of these studies to DLC analysis and design has been minimal [G2].

Virtually nothing is known regarding either the minimal amount of overhead required for data link control or the means of specifying DLC's to achieve minimal overhead. The only study [G5] dealing with these difficult questions did so under the following highly idealized conditions: arbitrarily large networks which are stable with regard to the number of users and network traffic, error-free transmission, failure-free data links, fixed routing, and infinite nodal buffers. Gallager [G5] demonstrated that the receiving node need be advised only of the starting time of message transmission and either the length of the message or the stopping time if synchronization on a network-wide basis is perfectly maintained. He also showed that as the allowable node-to-node delay increases, the required amount of link control overhead decreases, and suggested two (rather impractical) schemes for realizing this decrease in overhead.

The assumptions of error-free links, fixed routing, and perfect network-wide synchronization limit the utility of Gallager's study which, however, may prove to be a useful beginning for achieving

reduced overhead for network operation and management.

The current status of DLC's can be summarized as follows:

1. Existing data link controls provide for reliable node-to-node communication and are beginning to become standardized, but involve considerable overhead, (approximately one-third of the ARPANET traffic is DLC overhead in a heavily loaded network [K1].)
2. Knowledge regarding minimum required DLC overhead would be extremely useful, as would means for achieving minimum overhead.

V-3 Data Link Errors and Error Control

Disturbances which cause some bits to be changed during data link transmission necessitate adoption of measures to control such errors.

Errors are usually characterized either as random or as bursts [B2]. Forward error correction (FEC) codes have been developed to detect and/or correct both types of errors [L1, L2]. Normally, burst errors predominate and arise from switching transients, lightening, misaligned transmission equipment, maintenance disturbances and deep fades on microwave channels [B2]. Tests on Bell System channels indicate that the probability that more than m consecutive bits are in error in a block of $n \geq m$ bits is

$$P(\geq m, n) \approx a(m)n^X \quad (5-4)$$

Constant $a(m)$ increases as m increases. Constant $X \approx 1$ for $n \geq 10$, which implies $P(\geq m, n)$ in (5-4) varies linearly with n .

Although forward error correction is often considered and much discussed, most operating systems use an automatic-repeat-

request (ARQ) strategy. Data to be transmitted is segmented into k -bit blocks and $r=n-k$ check bits are then added; these check bits provide for detection of all but a fraction 2^{-r} of error patterns. Error detection results in a request by the receiver to the transmitter to retransmit the block containing the errors.

Various ARQ strategies are available [B2, S7, S8, B3, R2]. Stop-and-wait ARQ involves transmission of a data block, a wait by the transmitter for either a positive or negative acknowledgement (ACK) and either transmission of the next block or retransmission of the current block as required. Stop-and-wait ARQ stores only the current message in the transmitter's buffer, but involves considerable idle time, particularly when half-duplex lines which require modem turnaround delays, are used (see Section 3-2). Such cases favour the use of long block lengths. However the probability of a block error increases with the block length n , with the result that the number of retransmissions increases with n .

Continuous ARQ requires capability for simultaneous two-way transmission (full duplex), since transmission continues until a negative ACK (NACK) is received, at which time the message in error and all subsequent messages are retransmitted. Continuous ARQ is clearly preferable to stop-and-wait ARQ when both the block error rate and the time between transmission and receipt of the corresponding ACK are not too large. Continuous ARQ requires transmitter buffering of all unacknowledged messages, however.

A variation of continuous ARQ involves request for retransmission of only those messages in which errors have been detected. Such a strategy requires identification by the ACK of the message

in question. The result is a decrease in retransmission traffic but a small increase in packet overhead. The biggest penalty, however, is the increased destination buffer storage and operating difficulties which accompany the increase in probability of out-of-order packet arrivals, as explained in Sections 5-6, 5-7 and 5-8.

The relative throughput J for both stop-and-wait and continuous ARQ is as follows, where J is the average number of non-check bits received divided by the total number of transmitted channel bits (chips) including check bits and retransmitted bits $[B2, B3, S2, R2]$:

$$J_{sw} = k(1-P(n))/(n+D) \quad (5-5)$$

$$J_c = k(1-P(n))/(n+P(n)D) \quad (5-6)$$

In (5-5) and (5-6), k and n denote the number of information and information plus check bits, respectively, $P(n)$ the block error probability, and D the number of chips transmitted between the end of transmission of an n -bit block and the reception of the corresponding ACK. With $D=n$ (5-6) applies to the modified continuous ARQ strategy described in the previous paragraph. For small $P(n)$ ($P(n) \ll 10^{-3}$) throughput reductions result mainly from parity overhead for error detection, since retransmissions are infrequent in such cases.

A variation of the above continuous ARQ strategy involves transmission of positive ACK's only; when the ACK is received by the transmitter, the corresponding message or packet is discarded from the transmitter's buffer. If a positive ACK is not received during a time-out period R , the corresponding message is automatically retransmitted.

Selection of the appropriate value for R is, upon reflection, a rather complex matter. Too large a value of R requires a large buffer, since the number of packets retained at the transmitter increases linearly with R , and causes undue delay of those requiring eventual retransmission. Too small a value of R results in the retransmission and ultimate duplication of successfully transmitted blocks.

Sunshine [S5, S6] determined relationships between the average packet delay and R , as well as between throughput and R , and finally between delay and throughput. The results, shown for specific distributions of node-to-node transmission delays, indicate a linear increase in delay with R , with the rate of increase rising as the channel degrades. Throughput rises initially with R , and then levels off. For typical Erlangian delay distributions, graphs of throughput-vs-delay suggest choosing R such that "most" packets will have been successfully transmitted in the absence of link errors [S5, S6]. For exponential distributions, the optimum R is not as clearly defined.

Several questions remain regarding the choice of R . Sunshine's results use packet loss probability rather than the more tractable and measurable link error probability to characterize errors. The variation of the optimum value of R with traffic level is not considered, although flow controls as well as nodal buffer limitations would provide ultimate limits on actual network traffic. A reasonable approach is to optimize R for maximum traffic levels. Lower traffic levels will result in improved network performance which however will not be as good as that obtainable

if R were also reduced with the traffic level.

In networks containing both terrestrial and satellite links, large link propagation time differences will dictate the use of different values of R on the two types of links.

Transmission of NACK's to the transmitting node following detection of packet errors by the receiving node would encourage larger values of time-out period R , since time-outs would occur only in those rare cases when either transmitted NACK's or transmitted packets were not received. The larger R would reduce duplicate transmissions, thereby reducing retransmission overheads and hence delay. Use of NACK's would increase traffic overhead and software costs, however.

The use of positive ACK's and retransmissions following a time-out period has been shown to be totally reliable in the absence of data link errors [S6], in the sense that packets always reach the destination node without duplication, provided that there is no restriction on the number of retransmissions permitted [S6]. On the other hand the possibilities of link errors guarantees that no DLC exists which can, with certainty, avoid both loss and duplication of packets [B1].

Data link error rate and time-out period R influence the choice of packet length. Reduction of packet overhead and number of ACK's favours long packets. An increased data link error rate favours short packets, since vulnerability to errors resulting in subsequent retransmissions increases with packet length. Reduced retransmission delays and nodal buffer storage favours short packet lengths. All of these results were confirmed by Chu [C2] who determined the optimum block (packet) length B , block overhead b , average

message length l , channel error probability and R . Actual results in the form of curves were obtained for exponentially distributed message lengths, and both random and burst error channels, for both stop-and-wait and continuous ARQ. The actual optimum B is rather broad. Results for selective ARQ are unavailable, although Chu's approach could be extended to deal with this case.

It is not clear that ARQ is better than FEC. The latter is easily implemented using shift register logic $[L1, L2]$. Correction of single-bit and selected multiple-bit errors, as well as burst errors is also relatively easy, although the burst error codes normally require long error-free guard spaces $[L1, L2]$. The throughput rate for FEC remains relatively insensitive to channel error rate, while the packet error probability increases with channel error rate.

ARQ systems involve error detection only, which is easily implemented and, as noted earlier, leaves few (2^{-r}) errors undetected, where r is the number of check bits. ARQ adapts to channel deterioration by reducing throughput via retransmissions.

Simple FEC may be used in conjunction with ARQ on high error rate channels, in order to avoid large throughput degradations which accompany high retransmission probabilities $[B2, B3, S7, S8]$. An alternative is to reduce data transmission of the data link modem, as explained in Section 3-2.

When a feedback channel is available, implementation costs and error performance would seem to favour ARQ, although a convincing demonstration of its superiority over FEC is unavailable. Such a demonstration should include ARQ nodal buffer costs which would exceed those for FEC, since the latter need not retain packets for possible retransmission. These cost savings might be used to increase

data link capacity or reliability, and to offset error correction circuitry costs. Savings from software costs associated with generation of and accounting for ACK's would also be available for the improvement of systems using FEC.

Efforts to improve both FEC and ARQ strategies continue. Much of the FEC effort involves improved decoding of convolutional codes, which include Viterbi decoding simplifications. Convolutional codes, although powerful in terms of error correction, are of limited use in computer communications, since information bits are spread over many channel bit-sequences. Recent improvements in ARQ strategies are of limited applicability, since they involve multiple transmissions of messages, a strategy which is of interest only when the channel is prohibitively noisy [S7, S8].

V-4 Priority Queuing

The analysis in Chapter 3 on queuing, multiplexing and polling did not specifically consider the effects of message service priorities. In this section we consider priority schemes whereby all messages with priority p are served before any with priority $p-1, p-2, \dots, 1$, where p is an integer; $1 \leq p \leq P$. Our discussion here is a brief and limited sampling of a vast literature [K1, K4, J1, J2, B4] to which the reader is referred for further details. We have already considered the assignment of priorities by regulation of transmitter power in ALOAH networks (see Section 3-6) and have seen that the result is an overall increase in throughput.

Priority may be preemptive, in which case any message being served is discarded and returned to the queue immediately upon arrival of a higher priority message. In a nonpreemptive system,

service of a current message is not interrupted.

Assume that all messages in class i have Poisson arrival rates λ_i , mean length \bar{X}_i , and utilization $\rho_i = \lambda_i \bar{X}_i$. The rates λ , average length \bar{X} and utilization factor ρ for the collection of all messages is as follows:

$$\lambda = \sum_{i=1}^P \lambda_i \quad (5-7)$$

$$\bar{X} = \sum_{i=1}^P \frac{\lambda_i}{\lambda} \bar{X}_i \quad (5-8)$$

$$\rho = \lambda \bar{X}$$

$$= \sum_{i=1}^P \rho_i \quad (5-9)$$

Further, define

$$W_0 = \sum_{i=1}^P \lambda_i \bar{X}_i^2 / 2 \quad (5-10)$$

Analysis shows that the mean waiting time W_i for messages in priority class i is [K1]

$$W_i = \frac{W_0}{(1-\sigma_i)(1-\sigma_i + 1)} \quad (i=1,2,\dots,P) \quad (5-11)$$

$$\sigma_i = \sum_{j=i}^P \rho_j \quad (5-12)$$

where $\sigma_{p+1} = 0$.

As expected, W_1 does not depend upon the parameters of messages in lower priority groups, except through W_0 . The effect of any waiting which occurs while service of messages of lower priority is completed is included in W_0 .

Equation (5-11) can be used to assess the effect of assigning priority 2 to short messages and priority 1 to long messages (Thus $P=2$). Such a strategy would favour short messages, including acknowledgements and short control messages. In this case

$$W_2 = (\rho_1 \bar{X}_1 + \rho_2 \bar{X}_2) / (1 - \rho_2) \quad (5-13)$$

$$W_1 = (\rho_1 \bar{X}_1 + \rho_2 \bar{X}_2) / (1 - \rho_1 - \rho_2) \quad (5-14)$$

To be specific, let $\bar{X}_1 = 10$, $\bar{X}_2 = 1$ and $\lambda_1 = \lambda_2 = 0.5$. This situation might correspond to classes 1 and 2 including messages and ACK's respectively; each message (except those retransmitted) has an acknowledgement. Then $\rho_1 = 10 \rho_2 = 0.5$ and $\rho = 0.55$. Substitution into (5-13) and (5-14) shows $W_1 = 1.18$ and $W_2 = 0.53$. With no priority structure, $W = 0.92$.

Is the above priority scheme desirable? Certainly the ACK's wait less time at the expense of the messages whose waiting time is increased. However, the message waiting time is actually increased, since most messages do not require retransmission. In terms of reducing message delay fast ACK's are ineffective unless retransmission is needed. Since messages are retained in the transmitting node's buffer until the ACK is received, the above priority scheme will slightly reduce the required buffer storage, since the round trip time (excluding actual transmission time) is $W_1 + W_2 =$

1.71 vs $2W = 1.84$. The one-way transmission time is 1 and 0.1 for messages and ACK's, respectively.

In many situations, ACK's can be included as part of a message flowing in the same direction as the ACK. The ACK's overhead is thereby reduced, as is the overall traffic. If such a "piggyback" strategy is used, ACK overhead would be reduced by, say, 50 percent, in which case $\bar{X}_2 = 0.05$. Now $W_1 = 1.16$ and $W_2 \approx 0$ since $\bar{X} = 1.05$ and $\rho_1 = 1.05/2 = 0.55$. The message waiting time increases from 0.92 to 1.16 but no waiting is required for the ACK's. The delay T for messages increases from 1.92 for the no priority case to 2.21 for the "piggyback" case.

On the basis of the above assumptions, the piggyback scheme seems best. The non-priority scheme may be next best, since in practice it is actual message delay which is of interest.

If the short messages in the above discussion include other control information, such as routing update information, then the above priority scheme may actually reduce overall delay, since such information will provide for alternate paths which reduce message delays. Again, "piggybacking" of control information is desirable.

If short messages also include actual data packets (which might vary from a minimum to a maximum length), then the above two-class priority scheme reduces average message delay as noted in Section 3-3. Actually, short messages are favoured even in the absence of priority schemes, since these do not queue behind packets which are part of a long message [M5].

If a cost of waiting D_p is assigned to each priority class, it can be shown [K1] that the priority scheme described above can

minimize the total cost

$$D = \sum_{p=1}^P D_p N_p \quad (5-15)$$

where D is the total cost and $N_p = \lambda_p T_p$ is average number of p -priority messages awaiting service.

Equations similar to (5-7) to (5-12) can be obtained for preemptive priority disciplines. In this case, the highest priority class P shows values of W_p and T_p equal to those for no-priority schemes, except that λ , ρ , and \bar{X} are replaced by λ_p , ρ_p and \bar{X}_p , respectively.

The above discussion can be extended to include time-dependent priorities [K1]; for example the priority assigned to a message awaiting service might increase with the waiting time.

Priority queuing has been applied with considerable success to the evaluation computer job scheduling, for which various algorithms are available. The results are of interest in the present context, as indicated by a consideration of the round-robin (RR) scheduling algorithm, which is illustrated in Fig. 5-2. In RR scheduling, each job receives service for a time interval which is short in comparison with its length X , and is then placed at the end of the job queue. Analysis shows that both the waiting time $W(X)$ and time to complete service $T(X)$ for jobs of mean length \bar{X} and Poisson arrival rate λ is

$$W(X) = X\rho/(1-\rho) \quad (5-16)$$

$$T(X) = X/(1-\rho) \quad (5-17)$$

While these linear dependencies are expected in the absence of

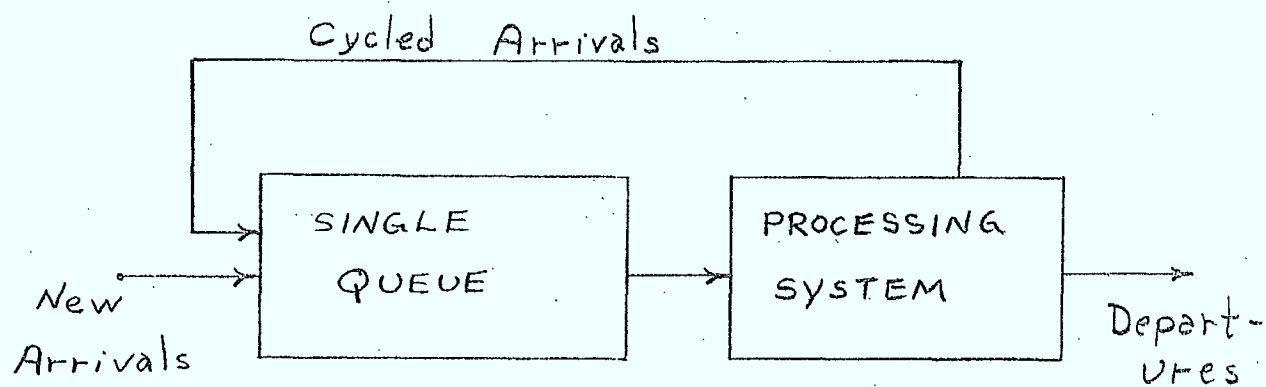


Fig. 5-2 Illustrating round-robin scheduling.

queuing, it is not obvious at the outset that they should apply when queuing for service is involved. The linear relationships are attractive; otherwise users would either combine many short jobs into one long one if T and W increased more slowly than X , or would break long jobs into many short jobs if the converse were true. The first strategy would make waiting times very long for users with inherently short jobs, while the second would add undesirable overhead.

The RR algorithm results provide a strong argument for breaking messages into packets much shorter than that of the average message. However, packet overhead and other considerations favour long packets.

Algorithms exist which favour long jobs over short ones, and conversely [K1]. These algorithms can be adapted in a rather obvious way to priority servicing of messages in computer communication networks.

We defer to Section 5-6 discussions concerning the application of priorities to flow control.

V-5 Adaptive Routing

Even though network flows λ_i and the multicommodity constituents are specified when the network is designed, the routes originally assigned will normally require modification during the course of actual network operation, for the following reasons:

1. Statistical fluctuations in traffic will occur; the original design is normally based on average flows.
2. Some links will experience prohibitively high noise levels and, in some instances, total failure.

3. Usage will vary over the course of any time period. In networks which span a wide geographical area traffic levels will vary in accordance with time-zone differences.
4. The traffic matrix $[r]$ will change over time as new users are added to the network and existing users change their usage patterns.

Routing may be either static, in which case routing remains unchanged unless the network is redesigned or modified, or completely dynamic, in which case routes are changed continuously. Between these two extremes lies quasi-static routing, in which changes in routes occur only at specified times and/or whenever extreme situations occur.

Routing may be either distributed in which case routing decisions may be made at individual nodes, or centralized in which case decisions are made at a central location. A mixture of distributed and centralized control is also possible.

Finally, routing algorithms may use either local or global information.

One would like to use a highly adaptive strategy based on global information. Unfortunately, physical realizability considerations, including limitations on the fraction of the total data traffic needed to transport status and other information, prevent realization of this ideal. The objective, then is to develop routing strategies which combine effective network operation with low data overhead.

Adaptive routing should improve network performance over that expected for fixed routing, since the latter does not fully exploit temporarily underutilized routes. Reliability would likely

increase, since noisy links would be avoided, particularly when alternate routes with low current utilization are available. Strong motivation therefore exists for devising efficient and effective adaptive routing algorithms. Although much effort has been devoted to the study and development of routing algorithms, an adequate understanding of the subject is as yet unavailable.

The following basic questions arise:

1. What fraction of routing control should be centralized, and what fraction distributed? The same question arises in other contexts, including corporate management and government.
2. What information should be used to generate routing decisions?
3. With what accuracy should information used to generate routing decisions be estimated? How often should the estimates be updated? Here we acknowledge that transmission of routing information reduces the network capacity available for actual data.
4. How should routing algorithms be assessed?

Other difficulties arise when adaptive rather than fixed routing is used. Packets may arrive out of order at the destinations causing buffer saturation and potential network deadlocks as explained in Section 5-7. Routing loops may develop, whereby messages circulate indefinitely among a set of interior network nodes.

Rudin [R3] presents an excellent study and discussion of various routing algorithms. He compares random routing, totally centralized routing, and delta routing which combines centralized and distributed routing control. Totally random routing [K5] involves no control and needs no control information, since messages

are routed to outgoing links selected at random. The messages exhibit random movement from node to node until they finally reach their destination. Random routing is very stable in the face of changing traffic patterns, but suffers from long delays which occur from the passage of messages through a large number of links [K5].

Centralized routing strategies are based on a matrix with entries $d_{ij}(t)$ which based on quantities $w_{ij}(t)$ and $r_{ij}(t)$ [R3]; these latter two quantities denote, respectively, the amount of information (number of message bits) at node i and destined for node j , and the capacity of the outgoing link from node i to node j . More generally $[W(t)]$ and $[R(t)]$ with entries $w_{ij}(t)$ and $r_{ij}(t)$ can be regarded as matrixes specifying the instantaneous work to be done by the network and the resources available to do the job. Quantities $d_{ij}(t)$ are obtained as follows [R3]:

$$d_{ij}(t) = kd_{ij}(t-\tau) + (1-k) [w_{ij}(t)/r_{ij}(t)] \quad (5-18)$$

Thus, d_{ij} depends on past estimates and a current update. Constants k and τ are to be optimized to tradeoff update frequency τ and network message throughput. Updating may be synchronous or asynchronous, the latter resulting in updates only when $d_{ij}(t)$ changes sufficiently. Asynchronous updates appear to result in better performance [R3].

The overall delay in sending a message from a current node i to final destination node j is estimated by adding the $d_{ij}(t)$'s for each link along a proposed path. If one path yields a considerably shorter delay, the appropriate outgoing link from node i is selected. Otherwise, outgoing link selection is based solely

on the outgoing link queues at the node in question. The shortest queue algorithm is a local algorithm whereby any given node uses a weighted average the waiting time on each outgoing link and each adjacent node's estimate of the time from the adjacent node in question to the final destination. The threshold for deciding whether or not to let the central control relinquish its routing decision to the local node is the parameter which specifies the relative amounts of central and local control. Rudin [R3] did not really address the important matter of computational time and complexity needed to perform the shortest path calculations, which are needed to make centralized routing decisions.

Analytical comparison of the various possibilities seem impossible. Simulation results on a limited number of networks indicate lower delay for delta routing although proportional routing, a centralized scheme whereby a given commodity is routed in split fashion over two outgoing links, is a close competitor in some situations [R3]. Comparisons of network message throughput are difficult to make; however, Rudin's results indicate some increase in throughput (or reduced required capacity for a given throughput) with optimized delta routing.

Attempts have been made to optimize algorithms which are solely distributed, in the sense that all routing decisions for traffic at any node are based solely on that node's routing table. Gallager [G6] provides a loop-free algorithm which minimizes average message delay. Basically, the approach is to increase flows on links for which the incremental delay is smallest, and to decrease flows on which the incremental delay is largest. The incremental delay on any outgoing link is obtained by differentiation

with respect to the flow along the link in question, and with respect to the network's offered traffic. The first term can be estimated from the formula for queuing delay by differentiation of (3-20) and by substitution of the current value of the link flow λ_i . The second term is obtained from information sent from nodes lying on the path between the destination node and the node in question. Unanswered is the question as to the amount of overhead information needed to operate the algorithm, and the algorithm's performance on operating networks.

A different approach to distributed routing optimization was proposed by Segall [S9] who used optimal control techniques to minimize the total time spent by messages in each nodal buffer. Thus, D is to be minimized where

$$D = \int_{t_0}^{t_f} \left[\sum_i \sum_{\substack{j \\ i \neq j}} X_i^j(t) \right] dt \quad (5-19)$$

where $X_i^j(t)$ is the total traffic at node i whose final destination is node j , t_f is such that $X(t_f)=0$ and D is the total delay over the time interval $[t_0, t_f]$. The sums are over all nodes in the network. Delay functional D is to be minimized with respect to the multicommodity flows u_{ik}^j , which denotes the flow along the link from node i to node k with j as the final destination, subject to the following constraints:

$$X_i^j(t) \geq 0 \quad (5-20)$$

$$u_{ik}^j \geq 0 \quad \text{all } i, k \quad (5-21)$$

$$\sum_j u_{ik}^j \leq C_{ik} \quad \text{all } i, k; j \neq i \quad (5-22)$$

where C_{ik} is the data link capacity between nodes i and k .

Generally, optimization involves solution of the two-point boundary value problem, which in the present context seems prohibitive in terms of time and cost, except when no traffic enters the network. This latter case is of minor practical interest. Segall's [S9] approach does have the advantage that assumptions regarding message length distributions and arrival statistics, including the independence assumption are not required.

Similar results for a variety of distributed algorithms were obtained [P1] for a tightly connected eight-node network as well as a 19-node ARPANET type of topology. Effects of node and link failures were also considered. The best algorithms were the ARPA type of adaptive algorithms described below and one using a time-dependent priority discipline to speed delivery of nodes which had been in the network for a long time.

Both centralized and distributed routing algorithms enjoy current use. The TYMNET network [S1, S10] used a centralized algorithm whereas ARPANET [K1, S1] used a distributed algorithm. The TYMNET control centre uses a version of shortest path routing between source and destination to determine the route for each user requesting access to the network. This route is maintained for the duration of the user's call. The ARPANET's routing tables at a given node are updated on the basis of the number of waiting messages on each outgoing link and the estimated delay at neighbouring nodes. Although ARPANET routing is completely distributed, it is nonetheless based on global information which slowly percolates throughout the network.

The demonstrated operational workability of both centralized

and distributed routing indicates the viability of each. The efficiency of the two approaches relative to the optimum is unanswered, as are the four questions posed at the beginning of this section.

Progress on improved routing strategies is needed and will undoubtedly occur. However specification of the optimum strategy is not likely to be soon forthcoming. In any event, hierarchical strategies should be emphasized for large networks, as explained in Section A-10 [K6, K7].

V-6 Flow Control

Congestion control as used here includes all those measures taken to prevent a network from being overloaded, or behaving as if overloaded. Flow control, an important aspect of congestion control, involves that collection of procedures used to prevent a user or group of users from hoarding network resources to the detriment of other users [R4]. Flow and congestion control is one of the most important, possibly the most important, aspects of network management. It is also one of the least understood aspects.

Section 3-5 dealing with contention multiplexing provides one illustration of the need for congestion control. Pure or slotted ALOAH systems which attempt carriage of an excessive aggregate traffic load provide unsatisfactory service to all users. Excessive traffic causes collisions between users to become so frequent that throughput falls and delay increases to prohibitive levels.

Congestion problems manifest themselves in other ways. Packets may be stored in filled destination buffers which cannot

accept those remaining packets needed to reassemble a multi-packet message which would, upon reassembly, be removed from the destination buffer. The needed packets stored in nodal buffers near the destination node are themselves prevented from moving toward the destination, with the result that these nodal buffers cannot accept additional traffic. Such a deadlock situation effectively disables some or all of the network [K1, K3, S1].

A reasonably effective flow control mechanism has been developed for a slotted ALOAH channel which services M users, each of which is permitted not more than one outstanding packet awaiting transmission [K1, L3]. The approach used involves definition of a threshold n_c such that when the number n of users wishing access to the channel is less than n_c the retransmission delay as described by parameter K in Section 3-5 assumes value $K=K_1$; when $n \geq n_c$ $K=K_2 > K_1$. The result of this control strategy is to increase the retransmission delay, and thereby reduce traffic offered as the number of active users increases. The choice of n_c itself is not critical; variation of throughput and delay about the optimum n_c is small [K1, L3]. The need for such control is motivated by studies [K1] which show that an uncontrolled infinite population always becomes unstable eventually, and a similar tendency is shown for finite M [K1]. It would be of interest to consider the above control strategy, or a modification thereof, perhaps to include more than two values of K , to include the situation where more than one message for each of the various users awaits transmission.

Polling of nodes by a centralized node avoids congestion. In centralized networks which use concentrators congestion is an

unlikely possibility. Thus, it is distributed networks which provide the flow and congestion control challenges. Various flow controls have been proposed, and some have been tested in the field or by simulation. Useful analysis of the various control strategies seems extremely difficult.

Virtually all flow control schemes involve either or both of the following features [D1, C3, K8, K9, P1, P2, P3, P4, S5, S6 K1, S1]:

1. A message or packet is permitted entry to a network only if sufficient buffer storage is available and has been allocated somewhere along the source-destination route.
2. A message or packet is rejected either at the outset or during transit if the number of buffers occupied in a portion or in all of the network exceeds some threshold.

In the ARPANET multi-packet messages are denied network entry until such time as a sufficient number of buffers have been allocated at the destination node of the communication subnetwork [K1, S1]. Further, no more than eight undelivered messages are permitted between any source-destination pair. Packets are assigned sequence numbers, and any packet outside the sequence window is rejected by the destination node. These precautions are implemented to prevent buffer lockup. Also imposed in the ARPANET is the limitation that not more than eight packets are permitted on the output queue of any link leaving an intermediate node [K1]. The TYMNET network limits the number of bits that an intermediate node will buffer for any given user [S1].

The guaranteed destination buffer allocation scheme of

ARPANET does not apply to single packet messages which are transmitted with no guarantee of destination buffer space being available [K1, S1]. However the probability of safe delivery is high since single packet messages need not await the arrival of other packets, but can be given immediately to the destination user.

Following multi-packet message reassembly at an ARPANET destination buffer, the released buffers are reserved for a time-out period R_e for another message from the same source. Following R_e sec. these buffers, if not utilized, are returned to the general reassembly pool.

The number of outstanding end-to-end messages and outgoing link packets in the ARPANET and number of nodal bits in the TYMNET are the important variables whose optimization has not really been addressed. A similar comment applies to the time-out period R_e , and to the maximum message length in terms of the number of packets as well as to the packet length itself.

Isarithmic (meaning constant number) congestion control has been proposed for use on the NPL network [D2, P3, S1]. The method requires each packet to first acquire a permit before being permitted network entry. Upon reaching its destination, the packet releases its permit which may either be acquired by an outgoing packet or, if the number of waiting permits exceeds a threshold, will move to other nodes until acquired. The total number of permits and the number of unacquired permits allowed to wait at any given node are the important variables.

The isarithmic flow control mechanism, which is perhaps the simplest of those seriously proposed, seems to work very well on small networks in the absence of other flow controls, and provides

some additional improvement when used in conjunction with other flow controls [P4]. On larger, hierarchical networks, lockup tendencies arising from lack of permits at certain nodes have been observed [P4]. This latter problem was overcome by reducing the traffic level with attendant and probably unacceptable delay and throughput degradations [P4].

Chou and Gerla [C3] have attempted to unify the multitude of existing and proposed congestion control algorithms in a meaningful and coherent manner by specifying the ways in which a particular flow control strategy implements the two control features listed earlier. A multitude of possibilities exist. Regarding item 1, buffers may be allocated for some or all classes of messages at one or more of the following places: source node, destination node, or intermediate nodes. The buffers may be dedicated to individual users, shared among the totality of users, or both. Buffers may be allocated via prior reservations or upon arrival of messages at nodes. Regarding overflow prevention (item 2), many possibilities again exist. Thus, limits can be placed on the queue length of each outgoing link, or on the total number of messages stored at any node, or on the totality of messages in the system. Both allocations and overflow limits may be either static, or may vary in accordance with message, traffic and network status. Clearly, the number of buffers for each type of allocation, as well as the numerical value of the overflow limits creates unlimited possibilities.

Chou and Gerla [C3] developed a general program for simulating various flow control strategies. They used their simulator to assess and compare the ARPANET and "window" protocols on small

networks; the window protocol involves definition of a window W packets wide such that packets arriving at a source and falling within the window are accepted, sequenced and transmitted to the destination user for reassembly [S5, S6, C3, C4]. Packets outside the window are discarded. The window size W regulates the level of network traffic, and can be optimized [S5, S6, C3]. Bell Canada's proposed Datapac Network uses the window concept. In both the ARPANET and window protocols, the number of packets queued for any outgoing link was limited to eight. Preliminary results [C3] seem to favour the window protocol, particularly if precautions are taken to prevent some users (those closest to the nodal buffers which saturate) from monopolizing the network. Much more study on other networks under varying traffic loads and failure situations is needed as a basis for firm conclusions.

If node-to-node ACK's are delayed until the message to be acknowledged is leaving the node from which the ACK is being sent, then this strategy, together with "windowing" further reduces nodal congestion [R1]. However the overall effect on network operation is unclear, since the retransmission time-out interval's variance and mean are now increased. Use of NACK's would probably prevent delay and throughput degradations (see Section 3-2) at the expense of some increase in DLC overhead.

Quantitative analysis of flow control has been attempted. Pennotti and Schwartz [P2] considered the tandem links shown in Fig. 5-3, which consists of link traffic with Poisson arrival rate λ_0 and external messages with mean lengths λ_i ($i=1, 2, \dots, M$). The average lengths of the exponentially distributed messages is μ_i . To permit analysis the simplifying and somewhat unrealistic

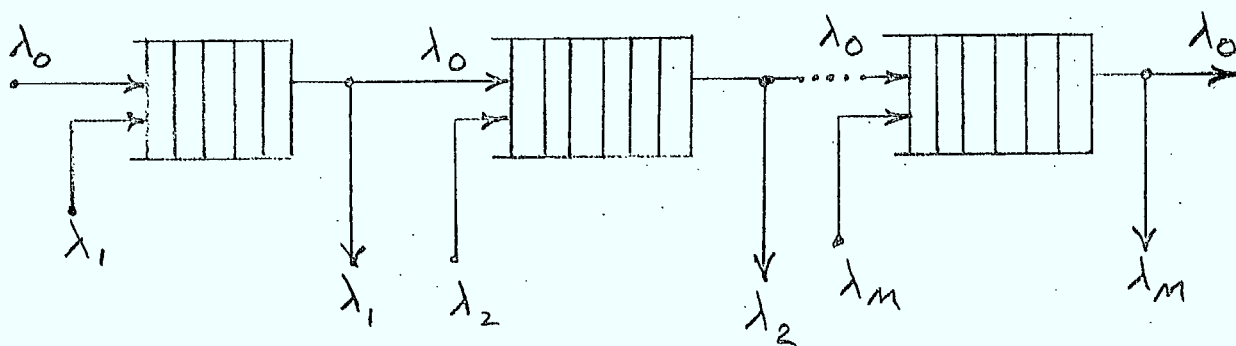


Fig. 5-3 Simplified model for flow control analysis

assumption was made that the external traffic entering and leaving the various points along the tandem path is equal. Using the above model, the relative increase in delay (defined as congestion) caused by the link traffic (λ_0) and seen by the external traffic λ_i was calculated under constraints on the maximum number of link messages N along the link, and again on the maximum number of link messages N_i ($i=1, 2, \dots, M$) at each node. Also calculated was the probability B that link messages seeking network entry are blocked by the congestion control strategy. Graphs showing congestion C vs B were obtained for both control strategies. Curves of C vs B for various link capacities and utilizations exhibited little difference between the two strategies. The above approach was extended by Chatterjee et al [C5] to include random routing over more than one path, and was used to assess the performance of a simple three-node, three-link network. A careful examination of the analysis indicates the considerable difficulty in applying it to an arbitrary network, notwithstanding the simplifying assumptions. Rather than being a criticism this comment indicates the difficulties inherent in flow and congestion control analysis. Simulation may be the only viable approach. In this regard, it seems that the rather general structure proposed by Chou and Gerla for considering various flow control strategies may prove very useful.

What is the optimum flow control strategy? No one seems to have developed a totally adequate approach to even deal with the question, which should probably include adaptive routing and priority queuing.

Flow control in conjunction with adaptive routing can be

very effective in preventing network congestion, since adaptive routing is effective if (and only if) some spare capacity exists for directing traffic from heavily loaded areas of the network [P4]. Kerr [K9] used a flow control scheme which limited the number of packets on any queue awaiting transmission to eight, and whose routing tables contained second and third choices for outgoing links which could be used if the more favoured links had full buffers. This simple myopic routing strategy resulted in a 50 percent increase in throughput and a slight improvement in delay at heavy simulated traffic loads. However, network lockup occurred; the problem was obviated by reconfiguring the higher level portion of the two-level hierarchical network [K9], which incurred increased link mileage.

Priority assignments can also control congestion; messages with low priority are denied entry to the network when the number of high priority messages exceeds a threshold. A study [P1] which compared various adaptive routing algorithms included some algorithms where all messages originally assigned the same priority received a higher priority after remaining in the network for a specified time. The resulting improvement in delay was rather impressive for a tightly-connected 8-node network, being on the order of 25 percent over an ARPANET routing strategy in a heavily loaded network. A small increase in throughput was also noted. Less impressive improvements resulted in a more sparsely connected 19-node ARPANET topology.

After careful study of both the references cited in this section and other references, it is easy to agree with those [P4, R4] who claim that the field of congestion control is at its

infancy, that a great deal of future effort is needed, and that this effort is essential to the smooth functioning of data communication networks.

V-7 Nodal Buffer Design and Management

Messages or packets arriving at internal (store and forward) network nodes must await service in nodal storage buffers. Packets arriving at destination nodes must be stored pending receipt of the remaining packets needed to assemble the complete message from the sender. Buffer capacity contributes to overall network costs, and should therefore be utilized efficiently. Actually, buffer costs should be optimized along with link flows, link capacities and topology during the network design phase.

Buffer considerations involve two separate but related issues:

1. How large a buffer is required?
2. How should the buffer be shared among outgoing lines at store-and-forward nodes, or between messages undergoing reassembly at a destination node?

Regarding the first item, existing literature [S1, K7, C7] suggests that buffer storage should be sufficient to limit the overflow probability P_0 to an acceptable minimum. Since a full buffer must discard new arrivals; the link throughput then falls by the factor $(1-P_0)$ [S1, K7, C6]. The difficulty is to specify the acceptable minimum. An alternative approach is to limit the buffer size such that the queuing delay encountered by the last message in a full buffer will be acceptable. However, use of priorities makes such delays difficult to calculate, and may indicate prohibitive buffer sizes if long delays are permitted to low

priority traffic. Adaptive routing further complicates delay calculations.

If messages with Poisson arrival rate λ enter a buffer serviced by a single outgoing link, then the average time that a message waits in the buffer is obtained from the queuing delay W in (3-6). The average number of buffer bits L is $[K1, S1]$

$$L = \lambda T \quad (5-23)$$

$$= T\mu C\rho \quad (5-24)$$

where C is the capacity of the buffer's output data link, μ^{-1} is the average message length and $\rho = \lambda/\mu C$.

The variance of L in (5-24) is obtained in an obvious way using the queuing results of Section 3-3.

The buffer overflow probability P_o can be estimated by assuming an infinite buffer, and using the probability that the length of the message queue L exceeds buffer size N . Thus,

$$P_o \approx \int_0^N f_L(u) du \quad (5-25)$$

where $f_L(u)$ is the probability distribution of L . For store and forward buffers, approximation of $f_L(u)$ by a Gamma distribution is indicated as explained in Section 3-3. The accuracy inherent in the above method of determining practical values of P_o ($\leq 10^{-3}$) seems excellent [S1, C6, C7, C8].

The requirement $P_o \leq 10^{-3}$ necessitates $N \gg \overline{L}$, where \overline{L} is the average amount of storage required. For typical length distributions with link utilization $\rho = 0.6$ and $\overline{L} \approx 40$, $N \approx 500$; for $\rho = 0.8$, $N \approx 850$ [C8].

To further reduce P_0 requires a modest increase in N , since P_0 decreases exponentially as N increases linearly [C6, C7, C8]. In the above example with $\rho \approx 0.6$ and 0.8 respectively; reduction of P_0 to 10^{-5} with $\overline{L} = 40$ requires that N increase to 900 and 1600, respectively [C7].

For exponential message lengths exact calculation of P_0 is possible [K1, S1, K7], as follows:

$$P_0 = \rho^N (1-\rho) / (1-\rho^{N+1}) \quad (5-26)$$

where ρ is the outgoing link utilization defined in Section 3-3. Exact calculation of P_0 for arbitrary message length is computationally tedious but possible using z-transforms [K1, S1, K7].

When nodes are serviced by two or more outgoing links, the buffer must be shared by these links. One sharing strategy involves completely partitioning (CP) the buffer such that each outgoing link is permanently assigned a fraction of the total buffer space; calculation of P_0 in this case is as explained for the case of a single outgoing link. The disadvantage of the CP strategy is that buffer space on some links will be available but not useable by one or more links with full buffers.

A second approach involves complete sharing (CS) of the buffer space among all outgoing links. This scheme suffers from the disadvantage that a subset of the outgoing links might use all of the buffer space, with the result that delay on these links would be large, while the uncongested links would be prevented from utilizing any of the buffer storage, thereby reducing network throughput. Calculation for P_0 and nodal delay T for this case is currently possible only for exponential queue lengths, and is

difficult even in this case [K7]. Simulation has been used [C7] to determine P_0 vs N for up to 10 outgoing links with varying link utilizations. Given N , P_0 is smallest for equal utilization and largest for all traffic on one link, as one would expect.

Various compromises between CP and CS exist [K7]. One involves limitation of the maximum length of any output link queue. Another involves assignment of a minimum amount of buffer storage to each link. A third involves a combination of these two strategies. The difficulty in specifying the maximum and/or minimum limits for the above schemes arises from the fact that expressions for P_0 and link delays are complex for exponential queue lengths [K7] and unavailable for other length distributions. Performance assessment and optimization of the various schemes using simulation is therefore indicated.

When there are two outgoing links, then the above three compromise schemes are identical and the situation simplifies considerably. Kamoun [K7] has compared delay (neglecting retransmission degradation) and throughput for the two-link case with equal flow and capacities on the two outgoing links and exponential message lengths. No one scheme is optimal over all traffic levels. Regarding throughput, low levels of traffic favour CS, high levels favour CP and intermediate levels favour the compromise scheme. Delay, excluding degradations due to retransmissions, and these are considerable for high traffic levels, seems to favour CS most and CP least.

Further studies of the various schemes are needed and should include more than two links, unequal output traffic, retransmission

effects on delay, and non-exponential message length distributions. A simulation approach would seem to be the only viable one.

The probability of overflow vs size of message reassembly buffers seems not to have been determined, although Sunshine [S5, S6] has considered the problem and obtained an expression for the probability $P(n)$ that a message packet arrives before one or more of its predecessors n or more packets earlier. The formulation is not directly suitable for dealing with the situation which occurs when a destination buffer is receiving packets for more than one message, with each message originating at any of the various source nodes. An analysis of this situation is needed, since destination buffer overflows may cause deadlocks or necessitate retransmission of many packets which have been correctly received. Such overflows can be avoided by flow control which, however, may seriously degrade throughput if overflow probability is not known. Overflow probability calculation is discussed further in the next section.

Another buffer management problem of some interest involves specification of buffer block size to compromise between overhead characters and unused characters at the end of a block due to the fact that messages will not exactly fill an integer number of buffer blocks, and buffer overhead. The problem has been considered and solved [S11].

Earlier papers which have dealt with dynamic buffering and block storage are listed as references [C9, P5, G7, C10].

V-8 Source Destination Controls

In many ways source-destination controls (SDC's) (sometimes called end-to-end controls) are similar to data link controls, since

both perform the functions of synchronization, error control, and initiation and termination of message delivery. Differences arise, however, because SDC's are concerned with the sending and receipt of messages, which may consist of many packets, whereas DLC's are concerned solely with node-to-node delivery of individual packets. SD synchronization involves ordering of the packets at the destination prior to delivery to the user. As noted earlier, out-of-order arrivals occur because of retransmissions, differing routes, or differing transmission delays.

As with DLC's considered earlier, existing SDC's operate reliability on several networks [D3, K1, S1, K2, K3, S6]. However, their efficiency is again in question, since neither the minimum required overhead nor ways of achieving this minimum are known. There appears to be not one study dealing with this question.

Sunshine's work [S6] on protocol analysis is of interest. He showed that in the absence of source-destination errors and source and destination failures, a protocol that rejects all packets which do not arrive in proper sequence at the destination never loses, duplicates or fails to deliver packets, and always delivers these in correct order. It is not difficult to see that excessive delays and low throughputs could result, however, from rejection of all out-of-order packets arrivals, as well as from stop-and-wait ARQ on an end-to-end basis. A viable approach is to define a window of width sufficient to permit most out of order packet arrivals to be buffered for subsequent placement in correct sequence, without incurring either an excessive destination buffer size or overflow probability [S5, S6]. As noted in Section 5-6, optimization of this window width has not been seriously considered, although Sunshine's

analysis [S5, S6] for single source-destination pairs is a useful attempt in this direction. The analysis does not show the effect of window width on either single packet or multi-packet message delay.

The actual assignment of sequence numbers to packets poses some difficulties. There must be enough numbers to ensure that an "old" undelivered packet with sequence number X from a specified source will not suddenly arrive and be regarded as the current packet number X . A large sequence number pool increases packet overhead. One possibility is to require the self-destruction of all packets which spend more than y sec. in the network, however such a scheme adds overhead and also faces us with the choice of y . The problem is not a serious one when flow control limits the number of outstanding messages m between any one source and destination. If the maximum message size is q packets per message, then the required number of sequence numbers is mq . In the ARPANET $m = q = 8$. Sequence numbers pose a much more serious problem in interprocess communications. (See Section 1-3).

Considerations pertaining to the selection of end-to-end retransmission time-out interval R_e are similar to those raised in Section 5-3 with regard to node-to-node time-out period R . Given the existence of end-to-end ACK's one might legitimately question the need for node-to-node ACK's (sometimes called hop-by-hop (HBH) ACK's). HBH ACK's permit larger values R_e with all the attendant advantages cited in Section 5-3. Gitman [G8] has considered the packet end-to-end (ETE) delay and throughput improvements due to incorporation of HBH ACK's. These improvements increase with the number of required link retransmissions as well as the number of

hops between source and destination.

It must be remembered that inclusion of HBH ACK's increases the network cost through increased message overhead, nodal buffer costs and software costs. If these costs were diverted instead into higher total network link capacity perhaps by adding more links, the overall effect might favour elimination of HBH ACK's for all but the most unreliable networks.

SDC protocol evaluation ultimately requires quantitative assessments of the effects of various protocols on message delay, error probability, and message costs, (including overheads and re-assembly buffer costs) vs message throughput. To actually calculate source-destination message delay from the individual link delays for the individual packets seems impossible. An alternative approach would be to assume a Gamma distribution for the end-to-end message delay, perhaps using different distributions for different types of messages such as interactive queries and file transfers, and to select the distribution's parameters using average number of hops, network traffic level, and message length distribution.

For a n -link path, the probability P_c of correct reception of a packet without end-to-end retransmission is:

$$P_c = \prod_{i=1}^n P_{c_i} \quad (5-27)$$

where P_{c_i} is the probability of correct reception by the node terminating link i . The effects of end-to-end retransmissions is obtained by summing a series, as in (3-29). The average P_c is obtained by averaging over all values of n weighted by the probability of n . The probability of a message error P_e is

$$P_e = 1 - (1 - P_c)^m \quad (5-28)$$

$$\simeq mP_c \quad (mP_c \ll 1) \quad (5-29)$$

where m is the number of packets per message. Actual calculation of P_c would be difficult; however an upper bound based on the maximum value of n would not.

Use of a Gamma distribution for end-to-end message delay would permit the calculation of reassembly buffer overflow, in the same way that node packet overflow is calculated as explained in Section 5-7.

Calculation of costs as given by message overhead would require knowledge of the packet retransmission probability as well as probability of end-to-end control packets. Measurement of overhead is an alternative. Such measurements indicate SDC overhead traffic as 25 percent of total traffic in a fully loaded ARPANET, most of which is destination-to-source requests for the next multi-packet message for which destination buffers are reserved [K1, K2, K3].

V-9 Protocols for Broadcast Networks

Extensions of EIS network facilities to remote locations and to mobile terminals is a function particularly suited for broadcast radio networks, since landlines may be either not feasible or uneconomic [K10, G9, R5].

Broadcast networks offer several advantages over point-to-point networks. Because physical links are absent, nodal repeaters are often easily moved to new geographical locations in response to changing traffic patterns. Explicit message acknowledgements are

unnecessary, since a node (node i) which has transmitted to a neighbouring node ($i+1$) need only wait for node $i+1$ to transmit the message to node $i+2$. This transmission to node $i+2$ is normally received by node i and regarded by node i as an ACK. If a particular node fails, a transmission power increase of neighbouring nodes permits the failed node to be bypassed.

The broadcast feature creates problems additional to those inherent in networks with point-to-point links. The broadcast feature will cause collisions to occur when two or more nodes in close proximity transmit packets which overlap in time. The collision problem could be solved by partitioning the frequency band into channels which are assigned to avoid overlap. However, as argued in Chapter 3 such an assignment is not well suited to bursty EIS traffic. Many copies of a transmitted packet may be generated following each nodal transmission, with the result that multiple copies of a packet will arrive at the destination node. Packet proliferation may cause the network to become heavily loaded and possibly inoperable as packets circulate or attempt to circulate endlessly among nodes.

Special routing algorithms are needed to circumvent the above problems. Gitman, Van Slyke and Frank [G9] have proposed and qualitatively assessed three such algorithms in terms of resource utilization efficiency.

The first algorithm, a non-directional broadcast algorithm, involves non-directional transmission to neighbouring nodes which, however, accept and retransmit the packet only if that packet has not been received during the previous L sec.

A hierarchical routing algorithm uses a hierarchical tree

arrangement of nodes, similar to that proposed by Kamoun [K6, K7]. A unique source to destination path results, and passes through a central station. The algorithm uses network resources efficiently, and avoids generation of multiple packets. However, path length is not minimized, since all packets must pass through the station. All nodes must maintain information to implement routing, but such information is easily modified to accommodate new nodes.

A directional broadcast routing algorithm involves storage at each node of an N by l_i routing table, where N and l_i are respectively the number of nodes and the number of nodes able to hear node i . An entry a_{ik} indicates the distance (or delay) from node i to node k when using outgoing link j . Node j accepts and retransmits a packet from node i and destined for node k only if k is closer to j than to i . The algorithm minimizes path length but does permit multiple copies of packets. If a node is added to the network, all routing tables must be updated, which will create considerable update information particularly if many of the source/destination terminals are mobile.

Packet radio networks have yet to be built, and detailed studies regarding delays, throughput, reliability and cost are lacking. One study [B5] was conducted concerning the reliability under various routing strategies, including: tree routing, whereby a unique path which includes a station connects any two nodes; restrictive routing whereby repeaters are assigned to hierarchical levels and routing must be via repeaters on different levels; and adaptive routing whereby communication between any two repeaters which can "hear" each other is permitted. Reliability measures

used include: expected fraction of repeaters able to communicate with the station, probability that all operative repeaters can communicate with the station, and expected number of node pairs communicating. Variables included nodal failure probability, station placement and repeater power level.

The result, obtained via simulation, showed that use of adaptive routing, use of routing paths which do not necessarily include the station, decreasing nodal failure probabilities, and increasing repeater power levels (which in effect increases the number of links) all drastically increase the reliability. The author's main conclusion from this initial study is that reliability must be explicitly considered in designing packet radio nets.

Examination of the reliability measures used in the above study indicate positive correlations between these, as expected. Precise relationships between the various measures as well as the one proposed in Chapter 2 are difficult to articulate. It is clear that the latter measure will correlate positively with those in the above study [B5].

Radio network design is in its infancy and requires further study, which will involve careful considerations regarding broadcast accessing methods as discussed in Chapter 3. Some of the current interest in mobile radio cellular channel assignment schemes may be of interest [J3, A2, C11, S12]. Cellular assignments involve division of spatial regions into cells and assignment of channels to cells in such a way that channels can be reused by non-contiguous cells to provide for overall system efficiency. The field of mobile radio, like computer-communications, is

experiencing rapid growth and constant change [J3, S12].

V-10 References

- A1 N. Abramson and F.F. Kuo, Eds., Computer-Communication Networks. Englewood Cliffs, N.J.: Prentice-Hall, 1973.
- A2 L.G. Anderson, "A simulation study of some dynamic channel assignment algorithms in a high capacity mobile telecommunication system," IEEE Joint Trans. Commun. and Veh. Tech., p.p. 1294-1302, Nov. 1973.
- B1 D. Belsnes, "Single-message communication," IEEE Trans. Commun., vol. COM-24, pp. 190-194, Feb. 1976.
- B2 H.O. Burton and D.D. Sullivan, "Errors and error control," Proc. IEEE, vol. 60, pp. 1293-1301, Nov. 1972; also in [C1].
- B3 R.J. Benice and A.H. Frey, Jr., "An analysis of retransmission systems," IEEE Trans. Commun. Technol., vol. COM-12, pp. 135-145, Dec. 1964.
- B4 K.R. Balachandran, "Purchasing priorities in queues," Management Science, vol. 18, pp. 319-326, 1972.
- B5 M. Ball, R.M. Van Slyke, I. Gitman and H. Frank, "Reliability of packet switching broadcast radio networks," IEEE Trans. Ccts. Syst., vol. CAS-23, pp. 806-813, Dec. 1976.
- C1 W.W. Chu, Ed., Advances in Computer Communications. Dedham, Mass.: Artech House, 1976.
- C2 W. Chu, "Optimal message block size for computer communications with error detection and retransmission strategies," IEEE Trans. Commun., vol. COM-22, pp. 1516-1525, Oct. 1974; also in [C1].
- C3 W. Chou and M. Gerla, "A unified flow and congestion control model for packet networks," in Conf. Rec., Third Inter. Conf. Computer Commun., Toronto, Canada, Aug. 1976, pp. 475-482.
- C4 V.G. Cerf and R.E. Kahn, "A protocol for packet network intercommunication," IEEE Trans. Commun., vol. COM-22, pp. 637-648, May, 1974.
- C5 A. Chatterjee, N.D. Georganas and P.K. Verma, "Analysis of a packet-switched network with end-to-end congestion control and random routing," in Conf. Rec., Third Inter. Conf. Comput. Commun., Toronto, Canada, Aug. 1976, pp. 488-494.
- C6 W.W. Chu, "Asynchronous time-division multiplexing systems," in [A2], pp. 237-268.
- C7 W.W. Chu, "Buffer behaviour for mixed input traffic and single constant output rate," IEEE Trans. Commun., vol. COM-20, pp. 230-235, Apr. 1972.

- C8 W.W. Chu, "Demultiplexing considerations for statistical multiplexors," IEEE Trans. Commun., vol. COM-20, pp. 603-609, June 1972.
- C9 J.H. Chang, "An analysis of buffering techniques in teleprocessing systems," IEEE Trans. on Commun., vol. COM-20, Part II, pp. 619-629, June 1972.
- C10 W.W. Chu, "Dynamic buffer management for computer communications," in Proc. Third Data Commun. Symp., pp. 68-72, Nov. 1972; also in [C1].
- C11 D.C. Cox and D.O. Reudnick, "The behaviour of dynamic-channel-assignment mobile communications systems as a function of the numbers of radio channels," IEEE Trans. Commun., vol. COM-20, pp. 471-479, June 1972.
- D1 A.A.S. Danthine and E.C. Eschenauer, "Influence on packet node behaviour of the internode protocol," IEEE Trans. Commun., vol. COM-24, pp. 606-614, June 1976.
- D2 D.W. Davies, "The control of congestion in packet switched networks," IEEE Trans Commun., vol. COM-20, pp. 546-550, June 1972.
- D3 C. Deparis, A. Dvenki, M. Glen, J. Laws, G. LeMoli and K. Weaving, "The implementation of an end to end protocol by EIN centres: a survey and comparison," in Conf. Rec., Int. Conf. Comput. Commun., Toronto, Canada, Aug. 1976, pp. 351-360.
- G1 J.P. Gray, "Network services in systems network architecture," IEEE Trans. Commun., vol. COM-25, pp. 104-116, Jan. 1977.
- G2 J.P. Gray, "Line control procedures," Proc. IEEE, vol. 60, pp. 1301-1312, Nov. 1972, also in [C1].
- G3 M.G. Gouda and E.G. Manning, "Protocol Machines: a concise formal model and its automatic implementation," in Conf. Rec., Third Inter. Conf. Comput. Commun., Toronto, Canada, Aug. 1976, pp. 346-350.
- G4 S. Ginsburg, The Mathematical Theory of Context Free Languages. New York, N.Y.: McGraw-Hill, 1966.
- G5 R.G. Gallager, "Basic limits on protocol information in data communication networks," IEEE Trans. on Inform. Theory, vol. IT-22, pp. 385-399, July 1976.
- G6 R.G. Gallager, "A minimum delay routing algorithm using distributed computation," IEEE Trans. Commun., vol. COM-25, pp. 73-85, Jan. 1977.
- G7 D.P. Gaver, Jr. and P.A.W. Lewis, "Probability models for buffer storage allocation problems," J. ACM, vol. 18, pp. 186-198, Apr. 1971.

- G8 I. Gitman, "Comparison of hop-by-hop and end-to-end acknowledgement schemes in computer communication networks," IEEE Trans. Commun., vol. COM-24, pp. 1258-1262, Nov. 1976.
- G9 I. Gitman, R.M. Van Slyke and H. Frank, "Routing in packet-switching broadcast radio networks," IEEE Trans. Commun., vol. COM-24, pp. 926-931, Aug. 1976.
- H1 M. Hellman, "Error-detection in the presence of synchronization loss," IEEE Trans. Commun., vol. COM-23, pp. 538-539, May 1975.
- H2 F.R.A. Hopgood, Compiling Techniques. New York, N.Y.: American Elsevier, 1969,
- J1 N.K. Jaiswal, Priority Queues. New York, N.Y.: Academic Press, 1968.
- J2 J.R. Jackson, "Queues with dynamic priority discipline," Management Science, vol. 8, pp. 18-34, 1961.
- J3 W.C. Jakes, Jr., Microwave Mobile Communications. New York, N.Y.: Wiley, 1974, ch. 7.
- K1 L. Kleinrock, Queueing Systems, Vol. 2: Computer Applications. New York, N.Y.: Wiley, 1976.
- K2 L. Kleinrock, W.E. Naylor and H. Opderbeck, "A study of line overhead in the ARPANET," Commun. ACM, vol. 19, pp. 3-13, Jan. 1976.
- K3 L. Kleinrock and H. Opderbeck, "Throughput in the ARPANET-protocols and measurement," IEEE Trans. on Commun., vol. COM-25, pp. 95-104, Jan. 1977.
- K4 L. Kleinrock, "Scheduling, queueing, and delays in time-shared systems and computer networks," in [A1], ch. 4.
- K5 L. Kleinrock, Communication Nets: Stochastic Message Flow and Delay. New York, N.Y.: McGraw-Hill, 1964.
- K6 L. Kleinrock and F. Kamoun, "Hierarchical routing for large networks," Computer Networks, vol. 1, pp. 155-174, 1977.
- K7 F. Kamoun, "Design Considerations for Large Computer Communication Networks," UCLA Dept. of Computer Science Tech. Rept. UCLA-ENG-7642, University of California, Los Angeles, Calif., 1976.
- K8 R.E. Kahn and R.W. Crowther, "Flow control in a resource sharing computer network," IEEE Trans. Commun., vol. COM-20, pp. 539-546, June 1972; also in [C1].

- K9 I.H. Kerr, G.R.A. Gomberg, W.L. Price and C.M. Solomonidies, "A simulation study of routing and flow control problems in a hierarchically connected packet switching network," in Conf. Rec., Third Inter. Conf. Comput. Commun., Toronto, Canada, Aug. 1976, pp. 495-502.
- K10 R.E. Kahn, "The organization of computer resources into a packet radio network," IEEE Trans. Commun., vol. COM-25, pp. 169-178, Jan. 1977.
- L1 S. Lin, An Introduction to Error-Correcting Codes. Englewood Cliffs, N.J.: Prentice-Hall, 1970.
- L2 R.W. Lucky, J. Salz and E.J. Weldon, Jr., Principles of Data Communication. New York, N.Y.: McGraw-Hill, 1968.
- L3 S.S. Lam and L. Kleinrock, "Packet switching in a multi-access broadcast channel: dynamic control procedures," IEEE Trans. Commun., vol. COM-23, pp. 891-904, Sept. 1975.
- M1 J.L. Massey, "Optimum frame synchronization," IEEE Trans. Commun., vol. COM-20, pp. 115-119, Apr. 1972.
- M2 D. Mandelbaun, "Synchronization of codes by means of Kautz's Fibonacci encoding," IEEE Trans. Inform. Theory, vol. IT-18, pp. 281-285, Mar. 1972.
- M3 P.M. Merlin and D.J. Farber, "Reoverability of communication protocols-implications of a theoretical study," IEEE Trans. Commun., vol. COM-24, pp. 1036-1043, Sept. 1976.
- M4 P.M. Merlin, "A methodology for the design and implementation of communication protocols," IEEE Trans. Commun., vol. COM-24, pp. 614-622, June 1976.
- M5 M. McDonald and H. Rudin, "Note on inherent and imposed priorities in packet switching," IEEE Trans. Commun., vol. COM-22, pp. 1678-1681, Oct. 1974.
- P1 R.L. Pickholtz and C. McCoy, Jr., "Effects of a priority discipline in routing for packet-switched networks," IEEE Trans. Commun., vol. COM-24, pp. 506-516, May 1976.
- P2 M.C. Pennoti and M. Schwartz, "Congestion control in store and forward tandem links," IEEE Trans. Commun., vol. COM-23, pp. 1434-1443, Dec. 1975.
- P3 W.L. Price, "Simulation studies of an isarithmically controlled store and forward data communication network," IFIP Conf. Proc., Stockholm, Sweden, Aug. 1974, pp. 151-154; also in [C1].
- P4 L. Pouzin, "Flow control in data networks - methods and tools," in Conf. Rec., Third Inter. Conf. Comput. Commun., Toronto, Canada, Aug. 1976, pp. 467-474.

- P5 R.D. Pedersen and J.C. Shah, "Multiserver queue storage requirements with unpacked messages," *IEEE Trans. Commun.*, vol. COM-20, Part I, pp. 462-465, June 1972.
- R1 R.D. Rosner, R.H. Bittel and D.E. Brown, "A high throughput packet-switched network without message reassembly," *IEEE Trans. Commun.*, vol. COM-23, pp. 819-828, Aug. 1976.
- R2 M.P. Ristenbatt, "Alternatives in digital communication," *Proc. IEEE*, vol. 61, pp. 703-731, June 1973.
- R3 H. Rudin, "On routing and "delta routing": a taxonomy and performance comparison of techniques for packet-switched networks," *IEEE Trans. Commun.*, vol. COM-24, pp. 43-59, Jan. 1976.
- R4 H. Rudin, "Chairman's remarks: an introduction to flow control," in Conf. Rec., Third Inter. Conf. Comput. Commun., Toronto, Canada, Aug. 1976, pp. 463-466.
- R5 L.G. Roberts, "Extensions of packet communication technology to a hand held personal terminal," in AFIPS Conf. Proc., SJCC, vol. 40, Montvale, N.J.: AFIPS Press, 1972, pp. 295-298; also in [C1].
- S1 M. Schwartz, Computer-Communication Network Design and Analysis. Englewood Cliffs, N.J.: Prentice-Hall, 1977.
- S2 J.J. Stiffler, Theory of Synchronous Communications. Englewood Cliffs, N.J.: Prentice-Hall, 1971.
- S3 R.A. Scholtz and R.M. Storwick, "Block codes for statistical synchronization," *IEEE Trans. Inform. Theory*, vol. IT-16, pp. 432-438, July 1970.
- S4 G. Seguin, "On synchronizable binary cyclic codes," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 589-593, Sept. 1975.
- S5 C.A. Sunshine, "Efficiency of interprocess communication protocols for communication networks," *IEEE Trans. Commun.*, vol. COM-25, pp. 287-293, Feb. 1977.
- S6 C.A. Sunshine, "Interprocess Communication Protocols for Computer Networks," Stanford Univ. Digital Systems Laboratory Tech. Rept. 105, Stanford University, Stanford, Calif., Dec. 1975.
- S7 A.R.K. Sastry, "Improving automatic repeat-request (ARQ) performance on satellite channels under high error rate conditions," *IEEE Trans. Commun.*, vol. COM-23, pp. 436-439, Apr. 1975.
- S8 A.R.K. Sastry and L. Kanal, "Hybrid error control using retransmission and generalized burst trapping codes," *IEEE Trans. Commun.*, vol. COM-24, pp. 385-394, Apr. 1976.

- S9 A. Segall, "The modelling of adaptive routing in data-communication networks," IEEE Trans. Commun., vol. COM-25, pp. 85-95, Jan. 1977.
- S10 M. Schwartz, R.R. Boorstyn and R.L. Pickholtz, "Terminal-oriented computer-communication networks," Proc. IEEE, vol. 60, pp. 1408-1423, Nov. 1972; also in [C1].
- S11 G.D. Schultz, "A stochastic model for message assembly buffering with a comparison of block assignment strategies," J. ACM, vol. 19, p. 483, July 1972.
- S12 L. Schiff, "Random-access digital communication for mobile radio in a cellular environment," IEEE Trans. Commun., vol. COM-22, pp. 688-692, May 1974.
- T1 S. Tavares and M. Fukada, "Synchronization of a class of codes derived from cyclic codes," Inform. and Contr., vol. 16, pp. 153-166, Apr. 1970.

