MODELS FOR COMPUTER NETWORKS†

Leonard Kleinrock
Associate Professor
School of Engineering and Applied Science
University of California at Los Angeles
Los Angeles, California 90024

Abstract

The important task of predicting performance of computer networks is considered. In this initial approach, both mathematical and simulation models are described, and the results obtained are compared so as to identify their differences. Suggestions are made with regard to creating more sophisticated mathematical models which will predict more accurately the behavior of computer networks. The driving force which motivates this analysis is the experimental computer network currently being implemented through the efforts of the Advanced Research Projects Agency in the Department of Defense.

I. Introduction

Computer networks are not new. SAGE[1] was one of the first as was the American Airlines reservation system.[2] Numerous military nets have been created and, of course, there is the huge electronically-switched telephone system. Recently CDC announced their nationwide computer network.[3]

Typically, these networks are large and costly. Their performance and operational characteristics are rather difficult to predict in the design stage. It is the intent of this paper to suggest mathematical and simulation models of computer networks in an attempt to predict their characteristics. Results of an elementary nature are given for the simpler models. More sophisticated models are also described for which the analyses have yet to be carried out.

II. The ARPA Experimental Computer Network

The system from which we draw our examples is the Defense Department's Advanced Research Projects Agency (ARPA) Experimental Computer Network.[4] The concepts basic to this network were clearly stated in Reference 5 by L. Roberts of the Advanced Research Projects Agency who originally conceived this system. This network will provide store-and-forward communication paths between a set of 19‡ nodes in the continental United States. The computers located at each of the nodes are highly incompatible (e.g. S.D.S. 940, DEC PDP-10, IBM 360/67, UNIVAC 1108, GE 635, ILLIAC 4, TX-2, etc.), and one of the major challenges is to design a network in which this assortment of varied hardware and software systems can communicate and cooperate with each other. The principle motivation for creating this network is to provide to each of the computer research centers those special resources which have been created at the other centers. For example, Stanford Research Institute will provide the role of network librarian and will offer its sophisticated text editing capability for massaging this vast data base; University of Illinois will allow access to the extremely high parallel processing speeds of its ILLIAC 4; University of Utah will serve as a major graphics center for picture processing; University of California at Los Angeles will process network measurement data and compare these to simulation and analytically predicted results.

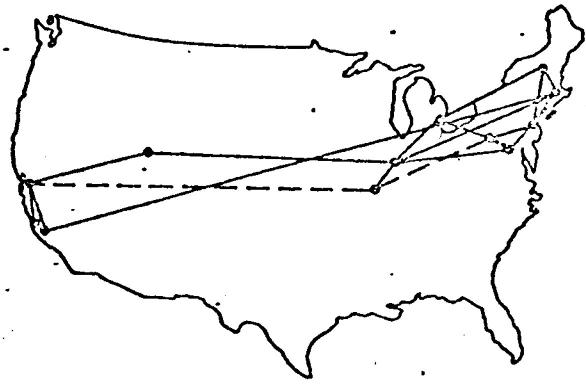
The example set of 19 nodes (mostly ARPA research contractors at universities) to be used in this paper is listed in Figure 1 and connections between these geographical centers is shown; two configurations are considered for purposes of this paper. Figure 2 shows these two configurations in a more readable topological form. Note that net 2 has three cross-country links, whereas net 1 has only two.

In order to interfere least with the existing operation of these various facilities, the message handling tasks (relay, acknowledgment, routing, buffering, etc.) will be carried out in a special purpose Interface Message Processor (IMP) colocated with the principal computer (denoted HOST computer) at each of the computer research centers. The communication channels will (in most cases) be 50 kilobit/sec fully duplex telephone lines, and only the IMPS (not the HOSTS) will be connected (through type 303 data sets) to these lines. This communication net, consisting of the lines, IMPS and data sets, serves as the store-and-forward system for the HOST computer net. Thus, for transmission between UCLA and UU, the direct path of store-and-forward transmission would pass through the UCLA HOST to the UCLA IMP to the SRI IMP to the UU IMP and then finally to the UU HOST, as shown in Figure 3.

---

† This work was supported by the Advanced Research Projects Agency of the Department of Defense (SD-184).

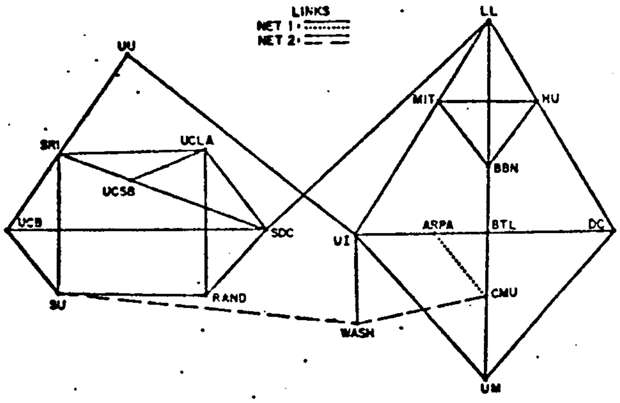
‡ This number, as well as the identity of some of the nodes, is subject to change at this stage of the planning.



Advanced Research Projects Agency, Washington, D.C. [ARPA]
Bell Telephone Laboratories, Murray Hill, N.J. [BTL]
Bolt, Beranek & Newman, Cambridge, Mass. [BBN]
Carnegie-Mellon University, Pittsburgh, Pa. [CMU]
Dartmouth College, Hanover, N.H. [DC]
Harvard University, Cambridge, Mass. [HU]
Lincoln Laboratories (MIT), Lexington, Mass. [LL]
Massachusetts Institute of Technology, Cambridge, Mass. [MIT]
Rand Corp., Santa Monica, Calif. [RAND]
Stanford Research Institute, Palo Alto, Calif. [SRI]
Stanford University, Palo Alto, Calif. [SU]
Systems Development Corp., Santa Monica, Calif. [SDC]
University of California, Berkeley, Calif. [UCB]
University of California at Los Angeles [UCLA]
University of California, Santa Barbara, Calif. [UCSB]
University of Illinois, Urbana, Ill. [UI]
University of Michigan, Ann Arbor, Mich. [UM]
University of Utah, Salt Lake City, Utah [UU]
Washington University, St. Louis, Mo. [WU]

TWO CONFIGURATIONS OF THE ARPA NETWORK
AND A LIST OF PARTICIPATING CENTERS

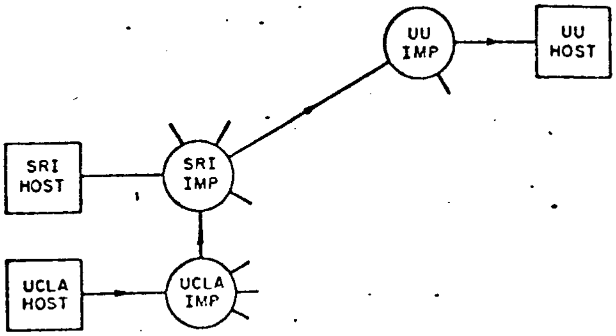
FIGURE 1



TOPOLOGICAL NETWORK DIAGRAM FOR
THE TWO CONFIGURATIONS

FIGURE 2

When the HOST has a message ready for transmission, it will be broken into a set of smaller packets (each of size approximately 1024 bits, or less) with appropriate header information. The IMP will accept up to eight of these (an assembly set) at a time. The packets will then individually make their way through the IMP network where the appropriate routing procedure will direct the traffic flow. For each IMP to IMP packet transmission, a positive acknowledgment is expected within a given time; absence of an acknowledgment (caused perhaps by channel noise detected by a cyclic error detecting code, or by lack of buffer space, etc.) will force the transmitting IMP to try the same or some different channel for retransmission. It is anticipated that an initial network of four nodes (SRI, UCLA, UCSB and UU) will be operational late this year, and that the full net will emerge sometime in 1970. One of the design goals is to achieve a response time less than 1/2 sec (round-trip) for short messages.



EXAMPLE OF TRANSMISSION FROM UCLA TO UU

FIGURE 3

The traffic matrix shown in Figure 4 gives the current (gross) estimate of traffic requirements between nodes (these numbers do not include the effect of acknowledgment traffic). We assume that the messages fall into two broad categories: one-packet short messages, typically averaging approximately 350 bits due to short commands and acknowledgments; and long, or multi-packet messages due to larger data transfers. In the subsequent results of Sections III and IV, we vary the mix of short and long (multi-packet) messages, maintaining an average input data rate to the entire net of 225 kilobits/sec.

ARPA NETWORK

From	To 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	Node Output Total	Node Input Total	I/O Rate Total
1. Dartmouth (DC)		2		1		2	1							1		1				8	4	12
2. M.I.T.	1		1	2	2	1	1	1	1	1	1	1	1	1	2	1	1	1	1	21	15	36
3. B.B.N.		1		1				1				1	2	1						7	12	19
4. Harvard (HU)		1	1		1							1		1				1		6	11	17
5. Lincoln Lab. (LL)		1				1	1		1	1		2		1			1	1		10	18	28
6. B.T.L.	2	1												1						4	8	12
7. Pentagon (ARPA)					1													1	1	3	10	13
8. Carnegie (CMU)		1	1						2			1		2	1		1	1		10	9	19
9. U. of Michigan (UM)		1			1			2		1	1			1		1			1	9	8	17
10. U. Illinois (UI)		2		1	5	1	1	1	1			20	1	2	2	10	1	1	1	50	13	63
11. Washing U. (WU)					1	1								1						3	5	8
12. U. Utah (UU)		1			5		1	2	1	8				1		5	1	1		26	29	55
13. U.C. Berkeley (UCB)		1	6	3		1			1					3	1	1	1	1	1	20	6	26
14. S.R.I.	1	1	2	1	1	1	1	1	1	1	1	1	2		2	1	1	1	1	21	19	40
15. Stanford U. (SU)		1						1						1						3	8	11
16. Santa Barbara (UCSB)				1						1	1	1					1			5	22	27
17. U.C.L.A.											1			1		1			1	4	10	14
18. RAND Corp.				1	1		2					1				1	1		1	8	10	18
19. S.D.C.		1	1				2							1			1	1		7	8	15
																			Total	225	225	

ASSUMED TRAFFIC MATRIX FOR ARPA NETWORK
(Entries in Kilobits/Sec.)

FIGURE 4

III. Mathematical Network Models

We present here some rather elementary mathematical models for computer networks along with some preliminary results. We construct our model so as to account for many of the salient features of the ARPA network described above although they clearly apply to more general systems as well.

In an earlier work on communication nets,[6] the author studied such nets using methods from queueing theory which provide an effective approach to these problems. We propose to use similar methods here for computer networks. Those characteristics which distinguish computer networks from those communication nets studied in Reference 6 include the following: (a) nodal storage capacity is finite and may be expected to fill occasionally; (b) channel and modem errors occur and cause repeated transmission; (c) acknowledgment messages increase the message traffic rates; (d) messages from HOST A to HOST B typically create return traffic (after some delay) from B to A; (e) nodal delays became important and comparable to channel transmission delays. Our elementary models will account for only a few of these, and as our models gain sophistication, more features will be included.

As in the study of communication nets, we assume that the message arrivals form a Poisson process with the average rates given in Figure 4 and also that message lengths are exponentially distributed with mean of 350 bits (note that we are, in some sense, only accounting for short messages and neglecting the multipacket traffic in this model). As justified in Reference 6 we make the independence assumption which allows a node-by-node analysis. We will also include features (c) and (e) from the previous paragraph, but neglect the others for now. We assume a fixed routing procedure (unique allowable path from origin to destination).

Considering line costs and termination costs (exclusive of IMPS and of course HOSTS), net 1 costs roughly $318,200 and net 2 costs $929,800. These costs assume a fixed channel capacity of 50 kilobit/sec on each line shown. In this case, as in Reference 6, we may calculate the average delay due to waiting for and transmitting over the i^{th} channel (say) as T_i where

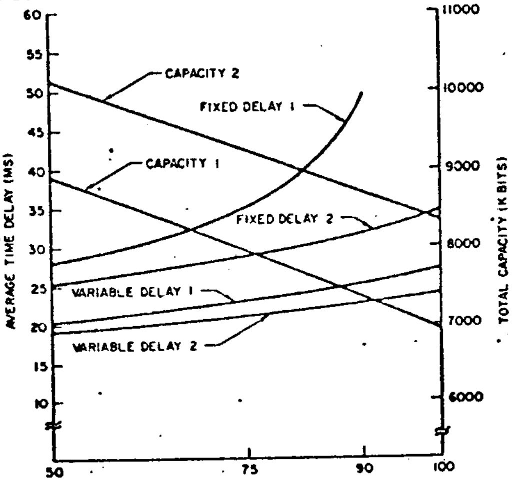
$$T_i = \frac{1}{\mu C_i - \lambda_i} \qquad (1)$$

where $1/\mu$ = 350 bits, C_i = 50 kilobits and λ_i = average message rate on channel i (as determined from the traffic matrix, the routing procedure and accounting for the effect of acknowledgment traffic). We may then calculate the delay T averaged over the entire network as

$$T = \sum_i \frac{\lambda_i}{\gamma} (T_i + 10^{-3}) \qquad (2)$$

where γ = total input data rate, and the term 10^{-3}= 1 millisec is included to account for the assumed (fixed) nodal processing time.

Carrying out this computation for nets 1 and 2, we get the average delay as a function of data rate shown as fixed delays 1 and 2 respectively in Figure 5. The data rate is adjusted by multiplying all entries in the traffic matrix by a constant between zero and one, where γ = 225 kilobits/sec at full data rate. The net and therefore its cost is held fixed in these computations. We see that net 2 is considerably more stable near full data rate as compared to net 1; this, of course, is due to the additional cross-country link.



TOTAL COST: CONFIGURATION 1 818.2 K$
CONFIGURATION 2 929.8 K$

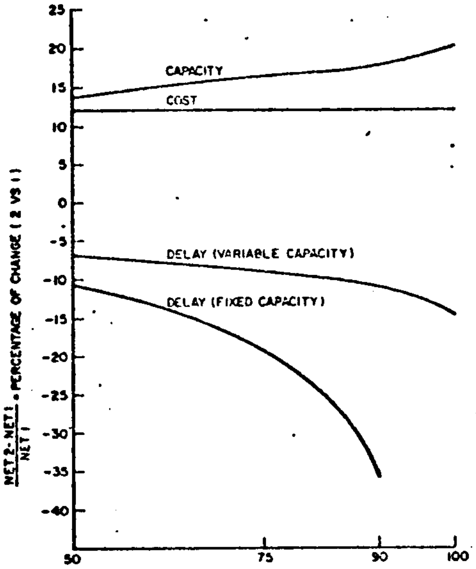
CAPACITY AND AVERAGE DELAY COMPUTED FROM MODEL AT FIXED COST

FIGURE 5

For theoretical purposes, we may allow ourselves the freedom of optimally assigning the channel capacity subject to the fixed cost constraint (as in [6]). The cost per channel is assumed to be a constant (termination charges) plus a linear function of capacity where the slope depends upon the length of the channel. From this calculation we obtain the optimum assignment (where we temporarily use a double subscript to indicate the channel from node i to node j):

$$C_{ij} = \frac{\lambda_{ij}}{\mu} + \frac{\left(D - \frac{1}{2} \sum_{m,n} \frac{\lambda_{mn}}{\mu} d_{mn}\right)}{d_{ij}/2} \left[\frac{\sqrt{\lambda_{ij} d_{ij}}}{\sum_{m,n} \sqrt{\lambda_{mn} d_{mn}}}\right] \qquad (3)$$

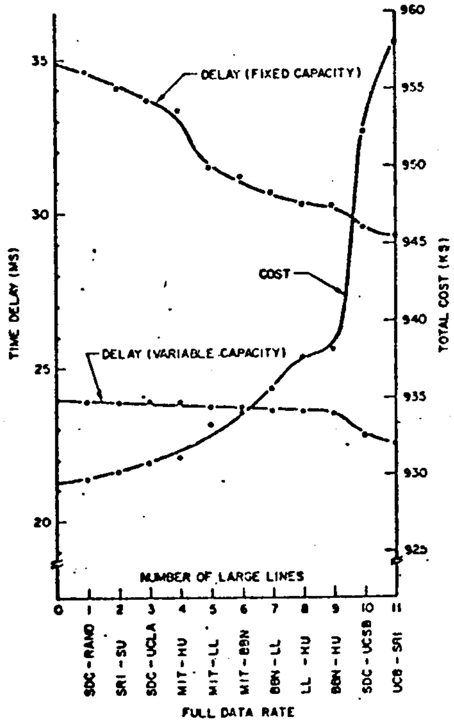
where $D = 477.3$ K\$ for net 1 and 579.6 K\$ for net 2 (the line termination costs are subtracted from the total net costs to yield D), $d_{ij} = .096 L_{ij}$ K\$ = cost per unit of capacity on the channel from node i to node j and where L_{ij} is the length (in miles) of this channel. The delay for channel i (we now revert to a single subscript again) is given by Equation (1). From this we may calculate T from Equation (2) for nets 1 and 2. These results are given in Figure 5 and are labeled variable delay. We note that the improvement obtained for net 1 is significant near full data rate; this is not as dramatic for net 2 since offered load did not especially tax this net's capability even with fixed capacities of 50 kilobit/sec. In this figure, we also show the fashion in which the total net capacity varies with data rate in the optimal assignment case; note that the capacity decreases linearly with increasing load. Figure 6 shows the percentage comparison between nets 1 and 2 for these variables. We see the importance of providing sufficient excess capacity for the high usage paths; for example, at only 90% of full data rate, an increase of approximately 12% in cost can reduce the average delay by roughly three times the cost increase (in percentage) for the 50 kilobit/sec nets.



PERCENTAGE VARIATION BETWEEN NETS 1 AND 2

FIGURE 6

In computing the optimal channel capacity assignment, we observe that certain of the channels require capacity greatly in excess of 50 kilobit/sec. Since the true physical net cannot support a continuum of capacities, we take the results of optimization and strongly quantize the capacities as follows: Beginning with 50 kilobit/sec channels in net 2, we identify the channel which would like the largest capacity (from the optimization); this is the SDC-RAND link. We then replace this with a 250 kilobit/sec channel and compute the average delay. Of course this increases the cost of the network; we therefore also calculate the average delay for a new optimized net with this new cost constraint. We then replace the next largest requirement with 250 kilobit/sec, etc. The results of this operation for the eleven most "needy" channels are shown in Figure 7. We observe that the optimized net improves slowly, whereas the fixed net improves in a more significant fashion. The cost increases rather slowly until the tenth large line is added; this is due to the fact that the SDC-UCSB distance is larger than the others. The trade-offs are clear.



EFFECT OF ADDING LARGE LINES OF 250 KILOBITS/SEC.

FIGURE 7

To generalize this simple model, we must account for item (d) at the beginning of this section, namely, the response from HOST B due to a request from HOST A. We suggest that an appropriate model is to recognize that HOST B is a time-shared computer system and model it as such. Numerous results are available which describe the behavior of the response time for time-shared systems.[7-11] The model we suggest therefore is that of a communication net whose destinations are time-shared systems; both the nets and time-shared systems have been modeled separately and we propose that the output of one feed the input of the other.

Further generalizations allow one to consider more general message length distributions by using the famous Pollaczek-Khinchin formula for the delay T_i of a channel with capacity C_i where the message length has mean $1/\mu$ bits and variance σ^2, where λ_i is the average message rate and $\rho_i = \lambda_i/\mu C_i$ as

$$T_i = \frac{2 - \rho(1-\mu^2\sigma^2)}{2(\mu C_i - \lambda_i)} \tag{4}$$

Note that for the exponential distribution, $\sigma^2 = 1/\mu^2$ and then Equation (4) reduces to Equation (1). We propose that this be used for T_i in Equation (2) for the average net delay T. We recognize by relaxing the assumption of an exponential distribution, that we are destroying the beautiful Markovian property of the traffic flow and thus our node-by-node analysis is incorrect; however, we offer this as a first approximation to the true behavior.

Regarding the buffer storage capacity in item (a), a most important consideration, we propose only a zeroth order approximation at this time. Specifically, one can create an infinite nodal storage model and then calculate the probability that more than a particular finite capacity was exceeded. What this fails to describe is the back-up effect of such buffer-induced blocking throughout the network. Models such as those constructed for sequential processing machines[12] may be helpful in this regard.

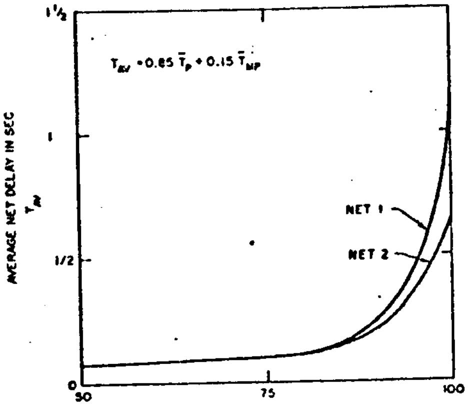
More elaborate models might include: general distributions for nodal delays; priority disciplines for message types at the nodes; recognition of the fact that response traffic is usually much greater in volume than request traffic; more detailed structure of delays and buffers within the nodes; alternate routing considerations.

IV. A Simulation Network Model

Since many of the mathematical models developed do not lend themselves to analysis, it is expedient to take advantage of the results obtainable from digital simulation methods. This we have done for the model described in Section II, and includes the five characteristics (a through e) listed in the second paragraph of Section III.

The network simulation program was written in G.P.S.S.[13] and runs on the IBM 360/75. It provides for the following variables: number of nodes; topology; channel capacity; traffic matrix and distribution of interarrival times; total data rate; nodal buffer size; modem and channel propagation delays; channel error rate; HOST→IMP transfer delays; short/long message mix with appropriate distributions at each node; IMP buffer level control; and IMP message management. The routing procedure is variable (e.g. fixed, adaptive alternate, random) however, only fixed routing algorithms are reported here. Delays for various message types and nodes, queue sizes, counts of nonacknowledged messages, channel utilizations, etc. are output variables whose statistics are measured by this program.

As in the mathematical analysis, the most significant overall performance measure is the average time T for a message to pass through the computer network from its origin to its destination. In Figure 8 we plot the simulation results for nets 1 and 2 with 15% of the total data rate in the form of multi-packet messages; shown is the average message delay as a function of data rate. In Figure 5 we plotted similar curves for the case of no multi-packet traffic; in both cases, we see the significant improvement of net 2 over net 1 at high data rates.

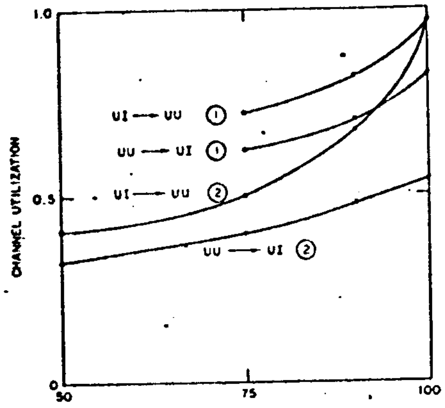


AVERAGE MESSAGE DELAY AS A FUNCTION OF DATA RATE FROM SIMULATION

FIGURE 8

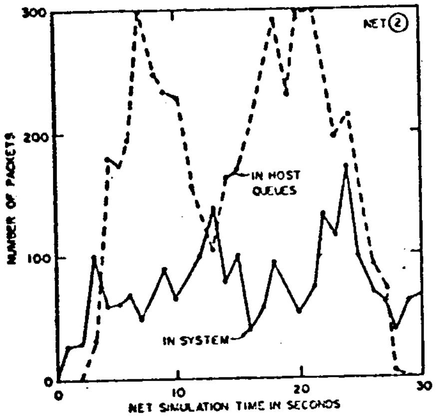
The most critical path in net 2 is from UI to UU; this is clearly seen from the traffic matrix in Figure 4 where the largest entry (by a factor of 2) is the UI to UU requirement. Since a channel exists between these two, the routing is direct. Figure 9 shows the channel utilization (fraction of time in use) for the two links UI to UU and UU to UI for both nets 1 and 2. We see that the utilization drops nicely for the UU to UI channel when the third cross-country link is added to net 1 to form net 2; however, we find at full data rate that the UI to UU channel is still running at an excessively high utilization for both nets. The effect of this high usage is to cause the large message delay times at full load shown in Figure 8.

One may observe the queue behavior as a function of time by studying Figure 10. We identify two packet storage locations: those waiting in the HOST for transmission to the IMP which are held up due to lack of storage room in the IMP



CHANNEL UTILIZATION FOR TWO CRITICAL CHANNELS

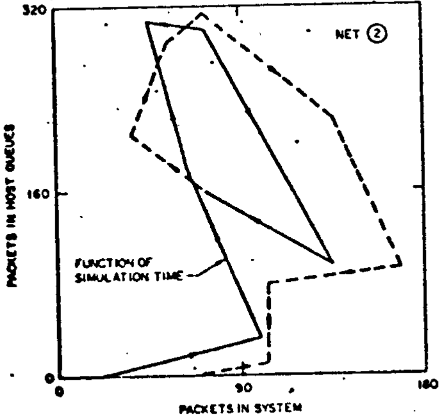
FIGURE 9



QUEUEING BEHAVIOR OF NET

FIGURE 10

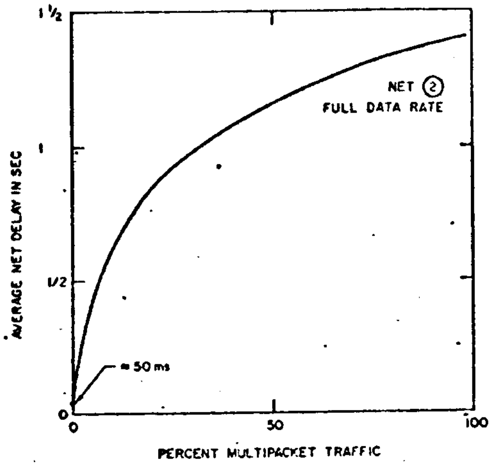
(note: only a fraction of the IMP storage is assumed to be available for this HOST to IMP traffic.— the rest is for IMP to IMP relay traffic); and those currently in the network in the process of making their way to their destination—these are said to be in "system." We observe a certain oscillatory behavior for these two curves which are out-of-phase with respect to each other. Figure 11 gives the cross-plot with time as the parameter; this more clearly shows the "see-saw" queueing effect. This behavior is due to HOSTS depositing many packets at once (from multi-packet messages) into the net when the IMP provides sufficient space; this shifts the queueing load from the HOST to the IMP. The IMP then blocks further transmission from the HOST until it discharges some of these packets. As a consequence when the IMP is emptying, the HOST tends to be filling, and vice versa.



CROSS-PLOT OF HOST QUEUES VERSUS SYSTEM QUEUES

FIGURE 11

Whereas all curves shown in this section were for a mix of 85% short and 15% long (multi-packet) messages, we now wish to examine the effect of changing this mix. The change in the average message delay is shown in Figure 12. The average data rate into the net was held constant at 225 kilobits/sec for this plot. The effect is rather dramatic: multi-packets increase the net delay significantly. The reason behind this is clear from a number of observations. First, we know that the increased variance inherent in the large range of multi-packet messages will increase queueing delays — see Equation (4). Also, increasing message lengths while reducing input message rates (i.e. keeping λ_i/μ constant to maintain constant data rates) will increase message delays in proportion to $1/\mu$ — see Equation (1). Moreover, the buffer blocking effect due to many packets entering simultaneously will inhibit IMP to IMP transmission until these buffers reduce their load. Other of our simulations indicate that these effects can be reduced for the short messages by giving them priority over the multi-packet traffic when the conflict arises in an IMP.



EFFECT OF CHANGING THE MIX OF MESSAGES

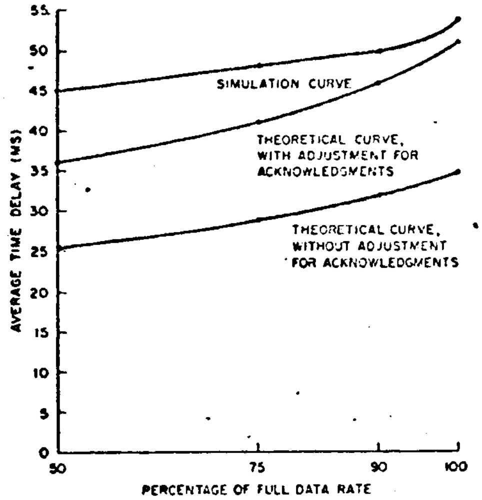
FIGURE 12

The finite nodal storage capacity appears to be one of the major causes of delay due to the blocking effect of full buffers. A thorough investigation of this behavior has yet to be carried out. Both analytic and simulation approaches to this problem would be useful.

It is important in any investigation of this sort to compare the results obtained analytically with those obtained through simulation. One comparison of this sort is given in Figure 13 where we plot the average message delay as a function of data rate both for the mathematical model analyzed in Section III and for the simulation. This theoretical curve is for an average message length $1/\mu = 350$ bits; however, this length has averaged the traffic due to acknowledgment messages. Since these acknowledgments should not be included in those messages whose average system delay is being calculated, we must increase the average length to be $1/\mu' = 560$ bits, but must continue to include the loading effects of such traffic; this we do by calculating

$$T_i' = \frac{1/\mu'}{C_i - (\lambda_i/\mu)}$$

to use in Equation (2) and, furthermore, adding 3 ms to T to account for a delay built into the simulation model to account for modem delays, etc. This theoretical curve with adjustments for acknowledgments is also shown in Figure 13. Considering the simplicity of the analytical model, it is reassuring that the results agree as well as they do.



COMPARISON OF THEORETICAL AND SIMULATION RESULTS FOR THE 50 KILOBIT/SEC. SYSTEM—NET 2

FIGURE 13

Further studies of alternate routing, channel error effects, priorities, response generated traffic, and storage size and control are yet to be undertaken.

V. Conclusions

We have attempted to develop some meaningful mathematical models and simulation models for computer networks, taking as our example the ARPA experimental network. Results from these models were presented, but such results must be viewed as highly preliminary and indicative of gross behavior at best. More sophisticated models have been proposed to improve upon these initial attempts. When the ARPA network commences operation, we plan to make meaningful measurements directly on the real net and compare these to our modeling efforts in an iterative way so as to evolve more accurate models. This last is a most crucial step in model building, and we plan to exploit the opportunity to its full extent. With such models available, we expect network users to construct network experiments which can be tested first by analysis and/or simulation before attempting any implementation of changes in the actual net.

Acknowledgment

The author takes great pleasure in acknowledging the assistance of Gary L. Fultz who created the network simulation program described in Section IV and of Kai-Min Chen who programmed the equations described in Section III.

References

1. R.R. Everett, C.A. Zraket, and H.D. Benington," SAGE: A Data Processing System for Air Defense," EJCC, pp. 148-155, 1957.
2. J. Evans, "Experience Gained from the American Airlines SABRE System Control Program," Proc. ACM National Meeting August 1967, pp. 77-83.
3. Business Week, February 8, 1969, p. 38.
4. P.A. Dickson, "ARPA Network will Represent Integration on a Large Scale," Electronics, Sept. 30, 1968, pp. 131-134.
5. L.G. Roberts, "Multiple Computer Networks and Intercomputer Communications," ACM Symposium on Operating Systems Principles, Gatlinburg, Tenn., Oct. 1967.
6. L. Kleinrock, Communication Nets; Stochastic Message Flow and Delay, McGraw-Hill, New York, 1964.
7. L. Kleinrock, "Certain Analytic Results for the Time-Shared Processors," Proc. of IFIP Congress '68, Edinburgh, Scotland, pp. D119-D125, August 5-10, 1968.
8. L. Kleinrock, "Distribution of Attained Service in Time-Shared Systems," J. of Computers and Systems Science, Vol. 3, pp. 287-298, October 1967.
9. B. Krishnamoorthi and R. Wood, "Time-Shared Computer Operations with Both Interarrival and Service Times Exponential," JACM, Vol. 13, No. 3, pp. 317-338, July 1966.

10. E.G. Coffman and L. Kleinrock, "Feedback Queueing Models for Time-Shared Systems," JACM, Vol. 15, No. 4, October 1968, pp. 549-576.

11. A.L. Scherr, "An Analysis of Time-Shared Computer Systems," MIT Research Monograph No. 36, 1967.

12. L. Kleinrock, "Sequential Processing Machines (SPM) Analyzed with a Queueing Theory Model," JACM, Vol. 113, No. 2, pp. 179-193, April 1966.

13. IBM Corporation, "General Purpose Systems Simulator III, User's Manual," Form H20-0163.