Proceedings of the 1986 Winter Simulation Conference
J. Wilson, J. Henriksen, S. Roberts (eds.)

ROUTING AS A FLOW CONTROL STRATEGY IN AN INTEGRATED CIRCUIT/PACKET SWITCHED COMMUNICATIONS NETWORK

Kenneth R. Hebert
U.S.A.F.
Udo W. Pooch
Department of Computer Science
Texas A&M University

ABSTRACT

This research addresses the analysis of an event-driven FORTRAN Simulation Model that simulates a special kind of Computer-Communication network. The network modeled has a circuit-switched communication subnet whose trunk lines carry both voice and data traffic simultaneously. This effort considers the viability of routing strategies as a mechanism for reducing congestion. The performance of seven alternative routing strategies are measured in terms of user-visible metrics.

Based on the experimental results obtained, this research concludes that fixed routing, the technique assumed by most models, is not as effective a tool for reducing congestions as would be a strategy based on link utilization. Minimization of congestion can be realized only if the routing strategy is adjusted as workload varies. Experimental data supporting these conclusions is presented.

1. INTRODUCTION

A computer-communication network may be described as an interconnected group of independent computer systems which communicate with one another (Chou 1983). The reasons for communicating are varied.

Sometimes a subtle distinction is made between a computer-communication network and a computer network. The difference between the two, according to Elovitz and Heitmeyer (1974), is that in a 'computer-communication network', the user is responsible for managing the computer resources; while in a 'computer network', the resources are managed automatically by a network operating system.

The emphasis of this paper is on the flow control of a network. In this regard, a computer network consists of a communications subnet (or backbone) together with the facilities needed to gain access to the subnet. The backbone of the network is comprised of communications processors (nodes) and trunk lines (links) which interconnect the nodes.

We present a methodology for flow control in a computer network. Section 2 provides some necessary background on the particular computer network being considered. We motivate an integrated circuit/packet-switched computer network and describe briefly a simulation model used to obtain performance data from such a network. Next we address the flow control design issues of an integrated network, and last present the results of applying the developed design methodology to various sets of integrated network specifications.

2. AN INTEGRATED CIRCUIT/PACKET-SWITCHED COMPUTER NETWORK

As used in this paper, the term "integrated circuit/packet-switched computer network" denotes a distributed computer network possessing a circuit-switched back-bone or subnet with numerous packet-switched local access networks feeding into the communications subnet.

2.1 The Case for an Integrated Network

Current military communication systems are generally designed to handle either voice calls or data transactions but not both. Such deployed systems use separate facilities for the two classes of traffic, thereby magnifying both the manpower and maintenance problems that already exist. The grade of service for these system is usually satisfactory, but crisis situations can and do force traffic flows that exceed system capabilities (Bially and McLaughlin 1980).

The concept of integrating voice and data rests on the fact that speech can be digitized and thus can be handled under packet switching schemes. Recent studies have addressed the problem of transmitting voice and data in the same computer-communication network. The scenario to accomplish such integration has been investigated; see Bially and McLaughlin (1980), Coviello and Lyons (1980), Coviello and Vena (1975), Dysart et. al. (1981), Hsieh et. al. (1978), Jenny et. al. (1975), Occhiogrosso et. al. (1977), Ozarow and DeRosa (1979), Rudin (1978), Takehiko and Shimasaki (1975), and Thurber (1978). Despite the many tradeoffs between packet switching and circuit switching, the consensus is that circuit switching delays have been improved to the point where both circuit switching and packet switching can be employed advantageously in the same network; Branscomb 1975, Hsieh et al. (1978) and Rudin (1978) emphasize that the integrated circuit/packet-switched network will become more prevalent in the future, eventually replacing circuit-switched or packet-switched systems.

2.2 Flow Control Strategies

From a user perspective, the computer-communications network is a "black box", to satisfy the needs for transmitting information. The network either satisfies the requirement to communicate within an acceptable time frame or it does not.

Experience with ARPANET and other packet switched networks has shown that flow control is a complex subject which can only be successfully addressed by a layered approach; see Andrews and Cooper (1978), Schwartz and Soad (1979), Thaker and Cain (1986), and Tymes (1981). These layers must work together in a synergistic fashion if problems are to be avoided. Gerla (1974) identifies the four levels of flow control.

The first level, node-to-node, involves the coordination of message traffic between two adjacent nodes of the communications subnet. Its objective is the prevention of buffer congestion and nodal deadlocks.

The second layer, source-node-to-destination-node, is an end-to-end approach to flow control. Its objective is to insure that traffic entering a source node is successfully transported to a destination node.

The third layer, host-to-node, deals with those controls necessary to insure that traffic is successfully transferred from the host (user) to the network access point (node). The final level, host-to-host, encompasses protocols between network users to insure the safe transmission of traffic.

Routing falls into the second category of flow control techniques. When used as a flow control strategy the objective of routing becomes the direction of traffic so congested links or nodes are avoided.

The classification scheme categorizes routing strategies into deterministic and stochastic algorithms.

Deterministic Algorithms. Deterministic algorithms derive routes based on a predefined set of static rules. This category, the simplest to implement, requires no information on the state of the network. Four basic strategies are described: Flooding, Fixed Techniques, Split Traffic and Ideal Observer.

Stochastic Algorithms. Stochastic algorithms operate on probabilistic decision rules rather than deterministic ones. Routes are selected based on an evaluation of network status. These techniques appear to have the greatest potential for accommodating network instabilities since they try to assess network status when determining routes. As a class, these techniques suffer from the overhead problems associated with exchanging status information on a timely basis. This becomes less of a problem with Common Channel Interswitch Signaling (CCIS) which provides for the constant exchange of network status and control information. The techniques involved with this procedure include (a) Random Technique; see Davies and Barber (1973) and Prosser (1962) and (b) Isolated Technique; see Boehm and Mobley (1969) and McQuillan (1977).

A great deal has been learned about the tradeoffs involved in devising a routing strategy. This paper extends these studies into the integrated environment where the added complexities of combined voice and data traffic make direct application of earlier results questionable. So far, research in the integrated environment has been limited to fixed routing strategies.

2.3 Network Environment

An integrated circuit/packet-switched network consists of the following major components:

A. Backbone Circuit Switch (CS) Nodes
B. Peripheral Packet Switch (PS) Nodes
C. Invariant Network Synchronous Time-Division-Multiplexed (TDM) Frame Switching Superstructure
D. Digital Network Using T1 Carriers and Digital Switching Nodes
E. Variable Subscriber Data Rates
F. Two Classes of Subscriber Traffic
    1. Class I: Real-time traffic that once started cannot be interrupted (voice, video, facsimile, and sensor).
    2. Class II: The general class of packet data, such as interactive, bulk, and narrative/message.

The backbone CS nodes and peripheral PS nodes form the nucleus of a distributed computer-communication network in which the transmission of data and voice between any two nodes on the subnet is accomplished by sharing the capacity of the T1 link. A Slotted Envelope Network (SENET) self synchronizing concept (Coviello and Vena 1975) is used to achieve simultaneous transfer of voice and data on the carrier. This concept treats the available bandwidth on a digital link as a resource for which all forms of communication must compete. Using SENET, the T1 link is synchronously clocked into frames of a fixed time duration, which are assumed invariant throughout the network. Each frame is partitioned into several data slots (channels) for which the various traffic types compete.

Voice (Class I) subscribers are terminated directly at circuit switches to avoid packetizing and any unnecessary routing overhead through packet switches. A Class I transaction results in a physical end-to-end connection for the duration of the call or a system loss (blocking) occurs. Although not a design requirement, packet switches are coallocated with the circuit switches. All data (Class II) subscribers are terminated at packet switches. While the Class II subscriber packet switch interface depends upon the individual terminal hardware configuration, the transmission of data between the packet and circuit switches is accomplished using TDM.

The regional routing doctrine for each packet switch, coupled with virtual switch connections, reduces overhead and the traffic congestion problem. As traffic is entered into a packet switch from subscriber terminals, it is queued for the relevant destination packet switch. A circuit switch connection is then initiated/terminated by the packet switch on behalf of this traffic. There are no explicit flow controls implemented in the model. Node-to-node controls become unnecessary because of the circuit switched backbone. Host-to-node controls are unnecessary because of the unlimited storage capacity of the PS nodes as are host-to-host controls. Finally, end-to-end flow control is realized via the routing strategy.

The number of incoming links is a function of the network configuration and can vary from one to the size of the network. Each incoming link possesses a fixed number of slots, based on its defined capacity. For model simplicity, it is assumed that the incoming links are also synchronized.

The circuit switch (CS) is the concentrator for all class I traffic, i.e. that requiring uninterrupted service. Class I service requests are assumed to have a poisson distribution and exponential interarrival times and lengths. Since the CS hub can only satisfy users when a slot is available, it can be viewed as having a capacity of L slots. The added constraint of limited server capacity, however, causes the value of L to vary from 0 to the total number of slots on the appropriate outgoing link. The value L is L = C - P, where C = the number of slots available on the proper link, P = the number of incoming packets to be immediately forwarded. The destination of each service request is a random variable with each network node having equal selection probability. The outgoing link requirement is determined by the routing strategy based on the destination and network status.

The Packet Switch (PS), is assumed to have an infinite storage capacity and is fed from the collection of all class II subscribers terminated at the current node. Transaction lengths (number of packets) are assumed to be geometrically distributed. The second source of class II traffic is the set of the incoming links. Depending on the routing philosophy selected, incoming class II packets are removed and queued to their appropriate destination along with nodal subscriber traffic. The queues are assumed to have infinite lengths. However, as mentioned above, the sorting mechanism is based on the packet's entry into the system. Thus, incoming packets could easily advance to the front of a queue if it entered the network prior to entries already queued. Entries from the Class II queue are accepted for service whenever there is available capacity in the appropriate outgoing SENET envelope.

3. CIRPAC SIMULATOR

3.1 Experimental Design

Given:
(1) a SENET-type integrated circuit/packet switched communications environment exists, and
(2) that this environment was initially optimized to a specific traffic flow pattern using progressive alternate (fixed) routing.

Question: can an alternative routing strategy be identified which reduces congestion as offered workload varies?

Congestion: Congestion in the integrated environment can be defined as a network condition where average message delay rise above a specified standard or a condition where blocking rises above a specified acceptable minimum.

3.2 The Simulation Model

The simulator is an event-oriented simulation with state changes occurring at zero-time events and the system clock advancing between these events. The permanent entities simulated include packet and circuit switching nodes and the T1 digital links connecting them. Temporary entities include voice calls and data packets. The original simulator was validated by Clabaugh (1979) and Kiemele (1984) via extensive sensitivity analysis.

For each possible combination of input parameters, the simulator is capable of generating a wide variety of performance measures. These include mean packet delay, average link utilization, packet throughput, average queue length, and fraction of calls blocked.

FLO, the driven of the simulator, is responsible for the three global management functions of model definition; initialization, simulation execution and experiment termination. At the highest level of abstraction, **FLO** is designed to accept as input a network configuration, a routing strategy, and the parameters needed to control the experimental run. **FLO** will then configure the network as requested and, applying the selected strategy, determine the largest workload which the network can accommodate without exceeding a specified grade of service. **DEFMOD** collects the information which defines the network configuration to be used for the experimental run. Flexibility is provided by allowing this data to take two forms, either a standard configuration definition or a snapshot from an earlier experimental run.

The second phase of the initialization process **DEFRTE**, is a subroutine responsible for gathering both the parameters which control the experimental run and the data needed to define and support the routing strategy used.

The final phase of the initialization process initializes the global variables and arrays used by **SIMULA**. This step is necessary because these variables/arrays will change over the course of a simulation run. **SIMINT** resets these to their appropriate values for incrementing the offered workload and then reinitializes the simulation for the next experimental cycle.

SIMULA realizes the simulation of the experiment defined by the previous three modules. It is an enhanced version of the simulator implemented by Clabaugh (1979) and it simulates the SENET-type environment.

The simulator is an extremely complex facility, comprising just over 5,000 lines of FORTRAN code (as opposed to 1800 for the original version). With this complexity comes a flexible capability to experiment with routing and flow control issues in an integrated environment.

4. EXPERIMENTAL RESULTS

4.1 Routing Strategies Used

The network simulation model was used to investigate generic routing strategies. The basic strategies and three of their variants were examined under progressively increasing workloads.

Fixed (RTPRBK and SSPRBK). The fixed routing strategy was implemented as the standard to be used as a norm of merit. **RTPRBK** selects a route by using the primary and secondary route from any source node to any destination node.

Traversing an indicated link will cause a transaction to reach either node Y or some intermediate node, I. If the identified link does not terminate at the desired destination, the link at coordinates (i,y) is investigated. Repeated examinations will eventually yield a link connected to the ultimate destination. At each stage, the available capacity of the selected link is examined. If there is insufficient capacity to accommodate the requested traffic, the link identified by an alternate path is examined for potential use. If no suitable link can be found, a bottleneck condition is returned in lieu of the requested route. Alternatively, if a route with sufficient capacity is identified a reference to that route is returned.

The fixed strategy, called "Progressive Alternate Routing" (Clabaugh 1979), is significantly more robust than fixed strategies which choose a path from one or more predefined alternatives. Since each link of a path can take on at least two values, the number of possible paths which are considered during route selection is at least 2^n where n is the number of links in a route.

SSPRBH is similar to **RTPRBK** except is limited to the first step of the route since this is the only element which will be used.

A variant of the progressive alternate routing technique, called "Primary-Only Routing" was also considered. Identified as **RTPRON** and **SSPRON** this variant was implemented by restricting the progressive alternate strategy to the primary routing table.

Random (RTRAND and SSRAND). Random route selection is made in the complete absence of information, including the desired destination. This strategy is possible in **FLO's** "single-step" mode of operation, where routes are re-evaluated at each intermediate node until the desired destination is reached.

Application of random routing concepts to the alternative philosophies afforded by **FLO** yields two variants of this basic strategy.

RTRAND represents the directed variant of random routing. when a request to traverse the network is received **RTRAND** randomly selects one of the routes which connect the desired source and destination nodes. Selection is based on a uniformly distributed random variable. All paths connecting the requested endpoints have an equal chance of being selected. Once selected, the route is evaluated to see if it possesses sufficient available capacity to handle the offered traffic. If there is, the index of the selected path is returned. It not, a variable is examined to see if an alternate route should be examined. This user-specified parameter, fixed at system initialization time, indicates the number of routes to be examined before a bottleneck condition is indicated.

SSRAND the second variant of random routing uses a true "drunkards walk". When asked to identify the "best route", or in this case, the "best next step", this procedure randomly selects one of the links emanating from the source node as the next link to traverse. **SSRAND** then ensures that there is sufficient available capacity on the selected link and, if there is, returns that "route". If there is insufficient capacity available, **SSRAND** determines whether alternative routes should be examined before a bottleneck condition is noted.

Adaptive Procedures. Adaptive procedures share a common approach to route selection. These strategies periodically examine network status information to guide the route selection process.

An important feature of any routing strategy is its ability to adapt to ever changing network conditions. This capability, called "adaptiveness", is a function of both the routing strategy under consideration and the routing philosophy in effect. While the former identifies the data and logic to be used in evaluating alternative paths, the latter determines the frequency with which route selections are re-examined. As the frequency of re-evaluations increases, so does the capability of the routing strategy to react to changes in network status.

Link utilization procedures (**RTUTIL** and **SSUTIL**) evaluate alternative paths based on the percentage of link capacity currently in use. **RTUTIL** responds to a service reques by identifying all paths which connect the desired source and destination nodes.

This procedure calls for the percentage utilization of each link along a candidate path to be computed and then raised to a user specified exponent. The resultant values are then summed to form a "metric" for that path. This metric forms the basis for path selection, selecting the path with the lowest metric.

This algorithm is varied slightly for situations where one of the component links of a path does not have sufficient available capacity to accommodate the requested workload. This situation is resolved by assigning that link an arbitrarily high value, one which is large enough so that any route using that link is unlikely to be selected.

The utilization metric reflects the relative line utilization of competing routes. Given two paths, both of length one, the route with the lowest utilization will be chosen. If two routes of unequal length are compared, the situation becomes more complex, with the shortest typically being selected. This rule of thumb fails, however, whenever the combined utilization of the longer route is less than that of the shorter one.

This weighting technique becomes even more complex when the effects of unequal link utilization along a path are considered. One could easily postulate examples where it would be imprudent to blindly select the path with the lowest utilization metric. This realization prompted the use of exponentiation to aid in path discrimination. Experimentally, it was determined that exponents greater than one (1) yield essentially the same relative values, therefore only one (1) and two (2) were used as exponents in the experimental runs supporting this analysis. These variants are identified as **UTIL-1** and **UTIL-2**.

The logic used by **SSUTIL** differs from its counterpart in two ways. First, as do all route selection procedures which support "single-step" operation, **SSUTIL** returns only the first step of the selected route. The second difference deals with the amount of data available to **SSUTIL** during route selection. Based on a user-provided variable CLOBAL, **SSUTIL** will adjust its "visibility". If GLOBAL is set to one (1), **SSUTIL** will select a route based on link utilization data for all remaining links between the current node and the desired destination. If GLOBAL is set to zero (0), **SSUTIL** will base its decision on only the status of links directly connected to the current node.

RTQCNT and **SSQCNT** base their evaluation of alternative routes on the size of the queues which have built up at intermediate nodes along a given path.

When a route request is received, **RTQCNT** first identifies all potential paths connecting the desired source and destination nodes. A queuing metric for each candidate path is then computed, and the path with the minimum queuing metric is selected.

Further complexity is added by eliminating the assumption of equal path length. As in the link utilization procedures, the queuing metric for each path is a summation of the individual link values. In general, this comparison technique favors the shorter paths, however examples can be constructed which cause the longer path to be selected.

The workings of **SSQCNT** differs from its counterpart by first, providing only the first "step" of the route selected, and second, allowing the restriction of its visibility.

The basis for analysis in the limiting procedures is also link utilization. (**RTLIMT** and **SSLIMT**). Rather than pursue the path with the lowest weighted utilization, these procedures select the shortest path such that all component links possess a progressively increasing quantum of excess capacity.

Again, **SSLIMT** is analogous to **RTLIMT**, differing from the latter in the ability to forward only the first step of the selected route, and the capacity to limit its visibility to only information available about directly connected nodes.

4.2 Resource Constraints

The flexibility of **FLO** provides the analyst with a virtually unlimited capability to investigate the integrated network environment. Not only can any network configuration be realized but given a particular configuration, the possible combinations of traffic arrival patterns/rates, routing philosophies and routing strategies are endless.

In this project, four major compromises were made to reduce the computing requirements. The first involves the use of 'generic' routing strategies to reduce the number of alternative strategies to be tested. A restriction of the workload settings was such that only balanced workload conditions were examined even though the arrival rate of the voice (Class I) and data (Class II) traffic can be independently varied.

Secondly, **FLO** was not allowed to iterate until it identified the largest workload which could be accommodated by a given routing strategy. Rather, a more economical approach was taken, whereby each strategy was examined under several smoothly increasing workload settings. Kiemele's sensitivity analysis (1984) demonstrated that the simulation model yields accurate results for workloads ranging from one to six Class I transactions per minute and from 100 to 600 Class II packet arrivals per second.

Finally, workload settings which provided only marginal information were eliminated. This resulted in the elimination of the first two workload settings for all strategies and the elimination of all workload settings for the random strategy. The result was a significant reduction in the required number of simulation runs.

The final compromise called for the experimental runs to be limited to only one routing philosophy, even though **FLO** makes it possible to examine three different philosophies. The philosophy examined was "end-to-end" routing, where routes are determined based on an evaluation at the source node and then remain constant for the duration of the voice call or message.

4.3 Experimental Data

In the data transmission world, one of the most readily visible metrics is packet delay. Two forms of this metric, the first, average packet delay, estimates the amount of time it will take a packet to traverse the network. The second metric the "aggravation factor", represents the percentage of packets which experience excessive delay.

Figure 1 presents the average packet delay (**APD**) statistics observed during the experimentation. The data is presented for each of the routing strategies examined.

The most available statistic for defining "excessive" is the one (1) second average packet delay criterion used by **CIRPAC** to optimize the network. Using this criterion as a basis for analysis, Figure 2 presents the percentage of packets taking longer than 1 second to traverse the network.

Blocking factor is defined as the percentage of class I service requests which are rejected due to insufficient resources along the path selected by the routing strategy under consideration. This traffic class prohibits queuing, therefore service requests cannot be stored until resources are freed. The analog of a class I rejection in the telephone network is a "busy signal".

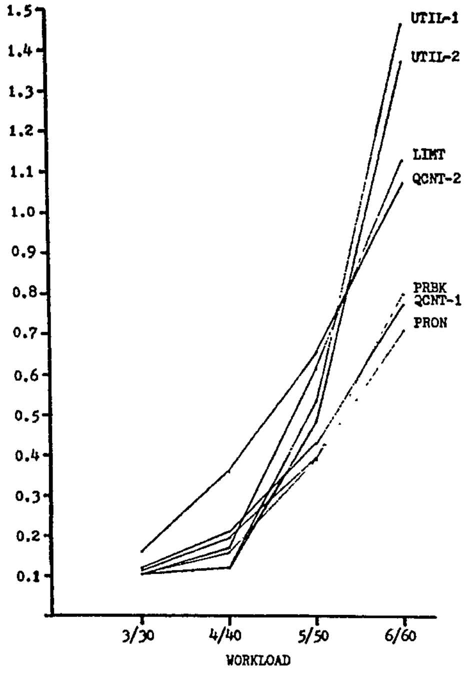

Figure 1: Average Packet Delay

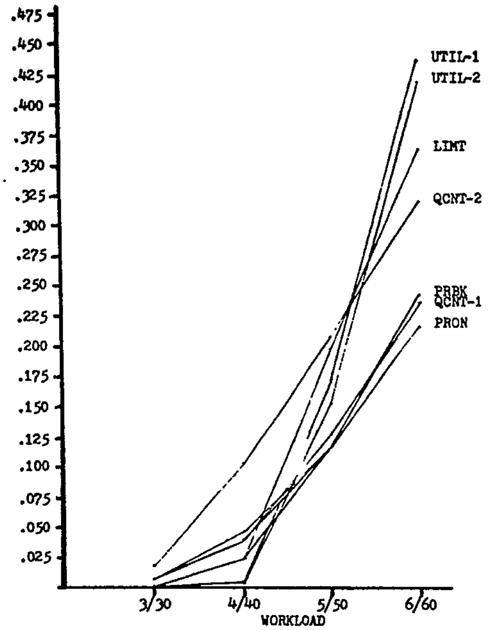

Figure 2: Percentage Delay > One Second

The optimum value for blocking factor is 0, i.e. no class I service requests are rejected. Such a figure is highly optimistic and either indicates a large reserve of communications facilities or exceptional management of resources. In practice, a communications network would be designed to keep the blocking factor below some threshold. The threshold used by **CIRPAC** in optimizing the network was 10%.

Figure 3 summarizes the blocking factors observed during the experimental runs. Review of the data clearly points out two facts. First, as would be expected, the blocking factor increases with traffic flow. Second, the observed data shows that an alternative strategy, specifically one based on utilization yields a significantly lower blocking factor.

The third "user-visible" metric, throughput, is a measure of the volume of traffic which successfully traverses the network over a set time period. The emphasis here is on the work "traverse" since the typical user only views the end-to-end results.

The throughput statistics collected during the experimental runs are presented in Figure 4. These statistics take two forms. The first, link throughput, must be cautiously interpreted science this figure reflects the average number of packets flowing through a node during a given time period. Though this value would rise with increased "end-to-end" throughput, it would also rise if the routing strategy under consideration selected extremely long routes.

The second form, message throughput, reflects the average number of messages (Class II transactions) which successfully traverse the network per second. Analysis of the data does not however, demonstrate significant differences between the alternative strategies.

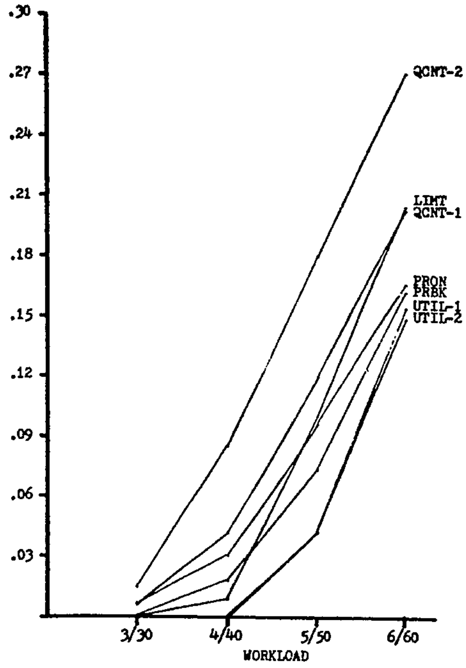

Figure 3. Blocking Factor

5. SUMMARY

From a user perspective, the details of how data and voice messages are transmitted are unimportant. The user is concerned only with two things: the cost to build the communications network and the performance it offers. These two features, unfortunately, are usually in direct conflict with the typical user striving to minimize cost while staying within an acceptable performance threshold. At some point, however, cost/benefit studies are completed, management decisions are made, and the network resources are acquired. From that point on, the user's only concern is end-to-end performance with performance typically being measured in terms of volume or response time. Common measures include: Throughput, Average message-delay, and Blocking.

These metrics provide the user with a quantitative assessment of how well the network satisfies his communications requirements. They are the ones he must contend with on a daily basis and are the ones most appropriate for measuring network performance.

Congestion in the integrated environment is thus being defined as a network condition where throughput and/or average message delay rise above a specified acceptable minimum. The specific values for the performance standards are established by management and form a key design constraint for the network.

The experimental results were examined from two perspectives. The first, proximity of observed metrics to "optimality", provides insight into how well each strategy performs with respect to a network blessed with unlimited communications resources. The second, proximity of observed metrics to that experienced by progressive alternative routing, provides a comparison of how well each strategy performs relative to the one for which the network was originally optimized.

The results clearly show that there are routing strategies which reduce congestion as workload is varied. Unfortunately, the results also show that none of the strategies investigated yield consistently superior performance over all workloads. Rather, it appears that the routing strategy should be adjusted as workload varies. Finally, the data indicates that a strategy based on link utilization out performs other strategies in most cases.

Based on a throughput analysis, all strategies performed essentially the same. If packet delay is the figure of merit to be used in judging network performance, the recommended routing strategy must change with workload.

Based on an analysis of the blocking characteristics demonstrated by the alternative strategies, **UTIL-2** is clearly the superior strategy. At all workload levels this strategy resulted in significantly reduced blocking.

Workload	PRBK	PRON	LIMT	QCNT-1	QCNT-2	UTIL-1	UTIL-2	
3/30	643.	640.	649.	651.	743.	644.	646.	Link Thru-put
4/40	868.	851.	967.	865.	982.	907.	907.	
5/50	1,113.	1,064.	1,344.	1,080.	1,232.	1,327.	1,319.	
6/60	1,347.	1,275.	1,611.	1,295.	1,483.	1,658.	1,669.	
3/30	2,686.821	2,686.013	2,688.358	2,685.397	2,687.376	2,688.358	2,688.358	Pkt Thru-put
4/40	3,580.856	3,582.875	3,583.154	3,966.497	3,579.925	3,582.536	3,582.297	
5/50	4,474.478	4,474.840	4,475.972	4,472.276	4,473.924	4,471.745	4,475.481	
6/60	5,357,443	5,357.671	5,360.960	5,365.313	5,359.349	5,362.365	5,363.899	

Figure 4: Throughput

REFERENCES

Andrews, F. B., Cooper, C. G. (1978). Probing NCR's distributed network architecture. *Data Communications*, 49-59.

Bially, T., and McLaughlin, A. J. (1980). Voice communications in integrated digital voice and data networks. *IEEE Trans. on Comm. Com-28*, **9**, 1478-1488.

Boehm, B. W., and Mobley, R.L. (1969). Adaptive routing techniques for distributed communications systems. *IEEE Transactions on Communications Com-17*, **3**, 340-349.

Branscomb, L. M. (1975). Trends and developments in computer/telecommunications technologies. *Proc. of the Organization for Economic Co-operation and Development Conference*, Paris, France, 57-77.

Chou, W. (1983). *Computer Communications, Volume I: Principles*. Prentice Hall, Inc., Englewood Cliffs,NJ.

Clabaugh, C. A. (1979). Analysis of flow behavior within an integrated computer-communication network. Ph.D. dissertation, Texas A&M University, College Station, TX.

Coviello, G. J. and Lyons, R. E. (1980). Conceptual approaches to switching in future military networks. *IEEE Trans. on Comm. Com-28*, **9**, 1491-1498.

Coviello, G. J. and Vena, D. A. (1975). Integration of circuit/packet switching by a SENET (Slotted Envelope Network) concept. *1975 Nat. Telecomm. Conf. (NTC-75)*, New Orleans, LA, 42-12 - 42-17.

Davies, D. W. and Barber, D. L. A. (1973). *Communications Networks for Computers*. John Wiley and Sons, London, England.

Dysart, H., Krone, M. and Fielding, J. (1981). Integrated voice/data private network planning. *IEEE 1981 Int. Conf. on Comm. (ICC-81)*, Denver, Co, 4.2.1 - 4.2.5.

Elovitz, H. S. and Heitmeyer, C. L. (1974). What is a computer network? *1974 Nat. Telecomm. Conf. (NTC-74)*, San Diego, CA, 1007-1014.

Gerla, M and Chou, W. Flow control strategies in packet switched computer networks. *1974 IEEE National Telecommunications Conference (NTC-74)*, San Diego, CA, 1032-1037.

Hsieh, W., Gitman, I. and Occhiogrosso, B. J. (1978). Design of hybrid-switched networks for voice and data. *IEEE 1978 Int. Conf. on Comm. (ICC-8)*, Toronto, Canada, 20.1.1 - 20.1.9.

Jenny, C. J., Kummerle, K. and Burge, H. (1975). Network nodes with integrated circuit/packet switching capabilities. IBM Research Report RZ-720.

Kiemele, M. (1984). Adaptive topological configuration of an integrated circuit/packet-switched computer network. Ph.D. Dissertation, Texas A&M University, College Station, TX.

McQuillan, J. M. (1977). Routing algorithms for computer networks - a survey. *1977 IEEE National Telecommunications Conference (NTC-77)*.

Occhiogrosso, B. J., Gitman, I., Hsieh, W. and Frank, H. (1977). Performance analysis of integrated switching communications systems. *1977 Nat. Telecomm. Conf. (NTC-77)*, Los Angeles, CA, 12:4-1 - 12:4-13.

Ozarow, L. and DeRosa, J. (1979). A combined packet and circuit-switched processing satellite system. *IEEE 1979 Int. Conf. on Comm. (ICC-79)*, Boston, Ma, 24.5.1 - 24.5.5.

Pooch, U. W., Greene, W. H. and Moss, G. G. (1983). *Telecommunications and Networking.* Little, Brown, and Company, Boston, Ma.

Prosser, R. T. (1962). Routing procedures in communications networks - part I: Random procedures. *IRE Transactions on Communications Systems*, 322-329.

Rudin, H. (1978). Studies on the integration of circuit and packet switching. *IEEE 1978 Int. Conf. on Comm. (ICC-78)*, Toronto, Canada, 20.2.1 - 20.2.7.

Schwartz, M. and Soad, S. (1979). Analysis of congestion control techniques in computer communication networks. *Proceedings on Flow Control in Computer Networks*, IFIP, North-Holland, 113-130.

Takehiko, Y. and Shimasaki, N. (1975). A study of future integrated service digital networks. *1975 Nat. Telecomm. Conf. (NTC-75)*, New Orleans, LA, 7-1 - 7-6.

Thaker, G. H. and Cain, J. B. (1986). Interactions of routing and flow control. *IEEE Transactions on Communications (COM-34)*, **3**, 269-277.

Thurber, K. J. (1978). Circuit switching technology: a state-of-the-art survey. *IEEE Proc. of Comp. Comm. Networks: COMPCON 78*, Washington, DC, 116-124.

Tymes, L. R. W. (1981). Routing and flow control in tymnet. *IEEE Transactions on Communications (Com-29)*, **4**, 392-398.

AUTHORS' BIOGRAPHIES

CAPTAIN KENNETH R. HEBERT received his B.S. (Computer Science) and M.S. (Computer Science) from the University of Southwestern Louisiana and his Ph.D. in Computer Science from Texas A&M University (1986). He is currently assigned to Headquarters Air Force Space Command, Peterson AFB, Co.

Headquarters Air Force Space Command
Logistic Support/Acquisitions
Engineering Support
Peterson AFB, Colorado
(303) 554-2884

UDO W. POOCH - Ph.D. Theoretical Physics, University of Notre Dame (1969), B. S. Physics, UCLA (1963).

Dr. Pooch, a well known expert in simulation and computer science is presently a Professor of Computer Science at Texas A&M University. In addition to simulation his research interests include artificial intelligence, networks, computer graphics, operating systems and computer architecture. He is the author or co-author of several books including, *Simulation: Principles and Methods*, Little, Brown and Company, (1980). He has over 90 technical publications in various journals, proceedings and contributions to books. In addition to being a reviewer for a dozen technical journals, he has served as series editor or guest editor for various publications including CACM, COMPUTER, Man Machine Interaction, Computer Networks, SIGGRAPH, Modelling, HICSS Proceedings and Computer Information Systems. He is very active in several professional societies including ACM, IEEE, ASA, SCS, ORSA, APS and SIAM. Dr. Pooch is a past chairman of the IEEE Technical Committee on Simulation and has been an ACM and DPMA National Lecturer for five years. He has been general chairman or program chairman for the Winter Simulation Conference, HICSS, SIGGRAPH, Tampa Annual Simulation Symposium, and the AMSE Winter Symposium. He has been recipient of both the Texas A&M University Distinguished Teaching Award and the College of Engineering Teaching Award he is listed in American Men and Women of Science, Who's Who in the South and Southwest, Men of Achievements and Who's Who in Computer Science.

Udo W. Pooch
Computer Science Department
Texas A&M University
College Station, Texas 77843
(409) 845-5498