Mai 78 1 N 2/26

Distributed processing in a packet-switching node: performance analysis

W. Bux and P. Kühn

Institute of Switching and Data Technics
University of Stuttgart
W. Germany

K. Kümmerle

IBM Zurich Research Laboratory
8803 Rüschlikon
Switzerland

Abstract

In the context of an integrated network for circuit and packet switching, a new structure for a switching node was suggested. Connection management and node control functions on the one hand and packet processing/storing on the other are performed by two groups of modules. Considering both the high throughput requirement of about 1200 packets/sec/node and the modularity requirement, node configurations may exceed 30 packet-handling modules. The prime objective of the present paper is to analyze the throughput characteristics of the multi-mini configuration for packet handling and to identify the major parameters on which it depends. In particular, the mechanisms which lead to saturation, i.e., decreasing incremental throughput with increasing number of processor modules, or even throughput degradation are investigated. The results provide a fundamental insight into how the most relevant system and traffic parameters determine throughput as a function of the number of packet-handling modules. In particular it was shown that the proposed partitioning of the packet-handling function does indeed permit high throughput before saturation is encountered. Due to the complexity of the problem the analysis was performed by means of simulation techniques.

EUROCOMP, London, 9.5. - 11.5. 1978

1. INTRODUCTION

In the context of an integrated network for circuit and packet switching, a recent study [1] suggested a new structure for a switching node: in particular, a novel concept for achieving high throughput in a distributed processing environment. The key features of this node architecture are (Fig. 1):

(1) Connection management and node-control functions, communications-link access functions, and packet-handling/storing are performed by three distinct groups of functionally separated modules, for reasons of cost effectiveness.

(2) Packet handling/storing in the data phase are performed in a set of independent modules (each containing storage and a micro-processor), for reasons of: (i) cost; (ii) ease in changing the packet-handling capacity upon change of load requirements, and (iii) a very high degree of reliability.

Flexibility in the modular packet-handling system implies that modules can be added and deleted without further change to the modules themselves. This requires that the modules be as autonomous as possible, i.e., each packet-handling module should be able to exercise autonomous control over a certain group of packet-switched (PS) connections. This excludes all central control mechanisms at the cost of a possibly high number of internal messages per packet flowing between modules. Considering the high throughput requirement of about 1200 packets/sec/node in the peak hour (as projected by the Eurodata Study [2] for 1985), node configurations will consist of a large number of packet-handling modules provided the total traffic is handled in the PS mode. To overcome the saturation effect of such a multiprocessor configuration a novel control concept was suggested in [1]. This mechanism significantly reduces intermodule communication (overhead) and thus yields a throughput characteristic which was estimated to be nearly a linear function of the number of modules over the range required.

The present paper is a more detailed investigation of the scheme proposed in the work referenced. The prime objective is to analyze the throughput characteristic of the multi-mini configuration for packet handling and to identify the major parameters on which it depends, in particular to determine the limitations of the control concept mentioned above, i.e., to identify the area where throughput saturation or degradation occurs. Though obtained for a specific system, the significance of the results lies in the fact that they are typical of multiprocessor systems which will become more and more important in the future.

The next section provides background information in the sense that it summarizes the operation of the switching node, and explains the main features of the control concept; Section 3 discusses the parameters which determine the throughput characteristic, and contains a brief description of the simulation model; finally, Section 4 discusses results. The Appendix provides a more detailed view of the simulation model.

2. NODE OPERATION AND CONTROL MECHANISM

This section follows reference [1] and provides the background information which is necessary for the remainder of this paper.

2.1 Node Elements and Circuit Switching Operation

Figure 1 shows the configuration of the integrated switching node for circuit- and packet-switched traffic. First, we shall briefly describe its elements and their operation.

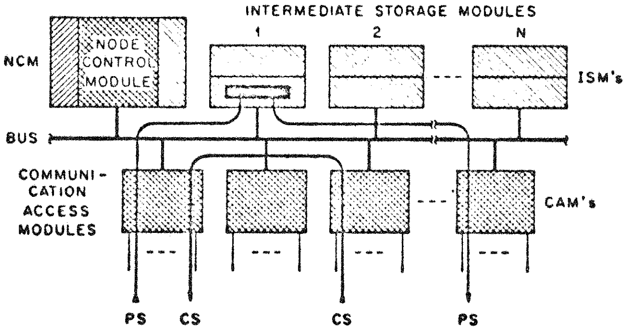

Fig. 1. Modular structure of switching node

Communications-link access module (CAM): A CAM interfaces a variety of communications links, such as multiplexed trunk lines of various speeds, nonmultiplexed lines, and local loops to the node. It contains the necessary logic and registers for exchanging data and control information with the other modules.

Node-control module (NCM): The NCM contains the software necessary to perform connection management and to monitor a connection. At the same time, it contains all node-control functions, the routing algorithms, as well as auxiliary functions for accounting, logging, etc. Also, the error control procedures of the NCM will be called upon to handle failures or other abnormal conditions in the other modules of the node which they are unable to manage by themselves.

Intermediate storage module (ISM): The ISM's provide buffer storage and processing capability for packet switching in the data phase. Each of these ISM's contains a small, very simple processor and a low-speed low-cost memory. Such a solution is suggested by the particular work-load characteristics: the many identical tasks lend themselves quite readily to parallel execution by identical programs in identical modules operating in parallel.

Information transfer bus (ITB): The ITB interconnects all modules. It employs a decentralized bus control with a daisy-chain scheme, a flexible priority structure, asynchronous bus communication and fixed-length blocks for the information transfer [3].

Distributing processing and control functions over a multiplicity of modules is a technique that has been employed to various degrees in other communications switching systems. Examples are the ARPA network's Pluribus 4], [5] or the Plessey System [6]. The modularity of these systems is well-tailored to the tasks they are supposed to fulfill. However, the environment of integrated circuit/packet-switched data networks places additional requirements on the flexibility of the switching node.

Referring to Fig. 1, the nodal operation in the circuit-switching (CS) mode will be briefly described. During the call set-up/disconnect phase the dialogue between user and node, besides the CAM's, primarily involves the NCM. At the end of the call set-up phase a real channel has been established through the network. This means for each node involved, that the registers of the CAM's concerned have been loaded with addresses in order to route the CS data flow through the node. It is an important feature of the solution that once the channel has been set up, the data flow is completely managed by the CAM's without intervention by the NCM.

The high degree of modularity leads to a range of intermodule communication problems. These problems appear to be most visible and best suited for analytical treatment when concentrating on the packet-handling processes in the ISM's. For the remainder of the paper we assume therefore that the total traffic is of the packet-switched variety.

2.2 Packet-Handling Mechanism

The objective of the proposed packet-handling mechanism is to reduce intermodule communications overhead in order to avoid the saturation effect typical of multiprocessor configurations. The basic idea of the new control scheme which was introduced in [1] is to split the packet-handling processes into two parts: one part where each process is disjoint from any other process in the system, and one part where the process must interact with other processes. Interaction is, at the very least, required for synchronizing the access to a departing trunk.

P and D functions: Figure 2 illustrates how the partitioned packet-handling processes are incorporated into the node structure already presented. The disjoint parts of the packet-handling process, i.e., the functions buffer allocation, packet read-in, packet storage, header processing, and acknowledgment handling, are represented by the function *P*. The process *dispatch* is represented by the function *D*. It is responsible for managing the outbound traffic for particular outbound lines, a supervisory, not a processing function. *P* and *D* functions communicate by means of node internal messages which are transferred by means of the ITB.

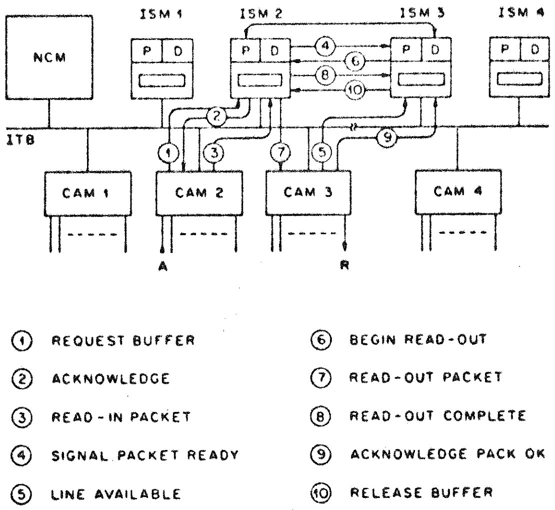

Fig. 2. Interactions of P and D functions in modular node structure

P and *D* functions are assigned in pairs for handling a particular packet-switched connection (virtual circuit). They are normally located in different ISM's. However, as each ISM handles a multitude of connections at a time, it is evident that each ISM must be capable of performing both functions, packet processing and controlling the dispatching, simultaneously.

CAM's "know" the addresses of all ISM's assigned to perform the *P* function for each of their incoming lines by means of registers where the information is stored during connection set-up. Once in the data phase, each packet can be routed by the CAM directly to the *P*-function ISM without requiring any supervisory action by the NCM. As will be shown later in detail, the ISM performing the *D* function is permanently associated with a particular CAM, whereas the assignment of an ISM for performing the *P* function changes dynamically depending on the load distribution in the node.

A schematic example will illustrate the function performed for handling a packet in the data phase by a transit node (Fig. 2): when the first character of a packet arrives at CAM 2 via trunk *A*, CAM 2 sends a request ① to ISM 2 to allocate the required buffer for that particular trunk. If the request can be satisfied, an acknowledgment ② containing the buffer address will be sent back to the originating CAM. Any further characters are immediately forwarded to ISM 2 to be read into the buffer ③. Once the entire packet has been assembled, ISM 2 performs error and sequence checks, analyzes the packet header in order to obtain the number of the outgoing trunk and, finally, signals a "ready to readout" ④ to ISM 3. Once trunk *R* is ready, CAM 3 sends a request ⑤ to ISM 3 which indicates that the next packet can be placed on the trunk. Subsequently, a request for packet ⑥ will be forwarded to ISM 2. The packet is sent from ISM 2 to CAM 3 character by character ⑦ and ISM 3 is informed of the last character transfer by a message ⑧. ISM 3 receives an acknowledgment ⑨ as soon as the packet is properly transferred to the next node. It then releases the buffer ⑩ in ISM 2.

Allocation of modules: The handling of a packet, once the modules constituting a cooperating pair have been appointed, has been explained. Here, the concepts which govern the formation of cooperating pairs will be discussed by means of some examples. The NCM originally assigns to every ISM the mode of associated ISM, i.e., to perform the *D* function with respect to a group of one or more CAM's. In the example represented in Fig. 3, the following associations are assumed (dashed lines): ISM 1 associated to CAM 1 and CAM 2; ISM 2 associated to CAM 3; and ISM *N* associated to CAM *M*. It should be pointed out that in the case of a full-duplex connection (*A-B*, *B'-A'*) as shown in the example of Fig. 3, one ISM may contain the *D* function for the "forward" direction of the connection, whereas another ISM contains the one for the "return" direction. The mode of associated ISM remains permanently assigned with respect to a group of CAM's and is only changed or canceled in the case of ISM failure or significant changes of the work load. Assignments are made on the basis of obtaining as equal a load distribution as possible.

Processing ISM's, i.e., ISM's performing the *P* function, are appointed for connections for which the node acts as network boundary node (source and destination) at the time a virtual connection is set up; for transit connections for low-speed trunk lines when the first character of a packet arrives, and for high-speed trunk lines when the trunk becomes active. Subsequently, we shall consider two examples for appointing an ISM as packet-processing ISM for a particular connection *A-B*.

The CAM handling the incoming line *A*, CAM 1 in Fig. 3, first sends a resource request block (= request for buffer) to the ISM associated to it. Provided the request can be satisfied, the address of the buffer area (page address) is sent back to the originating CAM. By virtue of this action ISM 1 is now assigned as the processing ISM for that particular connection. Since it has been assumed that the outgoing line *B* is attached to CAM *M*, ISM *N* is the associated partner in the cooperating pair. Figure 3 also shows that for the connection *B'–A'*, ISM *N* has been assigned as packet-processing ISM, whereas the *D* function is in ISM 1 and supervises the outgoing line *A'*.

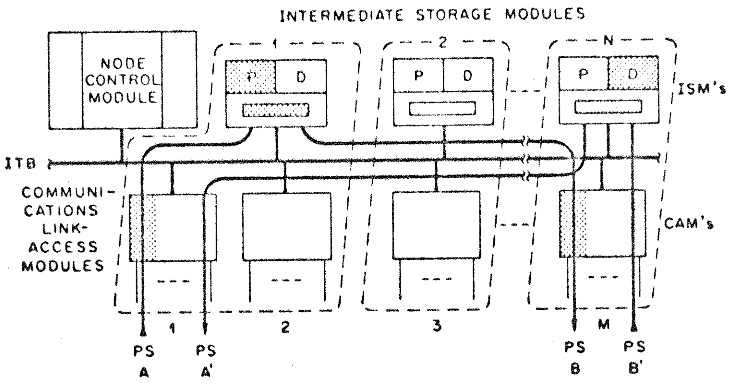
INTERMEDIATE STORAGE MODULES
1
2
N
NODE CONTROL MODULE
P
D
ISM's
ITB
COMMUNICATIONS LINK-ACCESS MODULES
CAM's
1
2
3
M
PS A
PS A'
PS B
PS B'

Fig. 3. Example for allocation of modules for connections A-B and A'-B'. Associations ISM-CAM are shown by dashed lines

Suppose the request for a processing ISM cannot be met, then, in the previous example, ISM 1 forwards the resource request block to ISM 2 or in other words the request *overflows* from ISM 1 to ISM 2. If the request can be satisfied there, as assumed in Fig. 4, ISM 2 is the packet-processing ISM for that particular incoming line. The appropriate buffer address is then sent to CAM 1 so that data can be sent to ISM 2 without involving ISM 1 any further. In the case that ISM 2 has no buffer available, the request overflows to ISM 3, etc. The policy of sending the resource request block to the respective associated ISM first prevents overloading of certain ISM's and distributes the load equally. The operation of forwarding requests which overflow to the next ISM is identical to a loop operation. We therefore denote it as overflow loop. Overflow activities beyond a certain threshold – they occur when congestion builds up in the node – will be prevented by a flow-control mechanism residing in the NCM. Note that the *D* function in ISM 1 maintains operations just as in the example of Fig. 3.

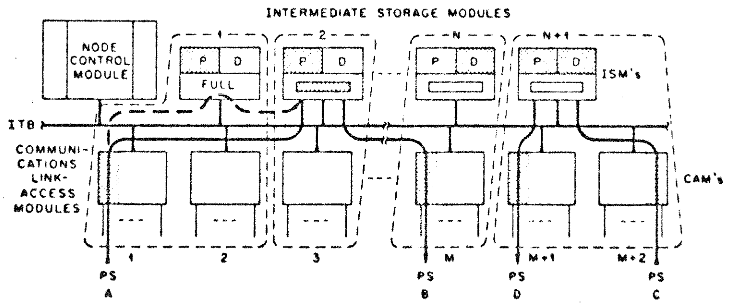

Fig. 4. ISM to ISM overflow

Finally, it may also occur that a connection involves a single ISM. This is shown in an example again in Fig. 4. Connection *C-D* is processed in ISM *N* + 1. As the same ISM also acts as associated ISM for CAM *M* + 1, it hosts both *P* function and *D* function for the same connection. Hence, interaction takes place within a single module.

3. GENERAL PERFORMANCE CONSIDERATIONS AND MAIN FEATURES OF MODEL

3.1 Sources of Throughput Degradation

The ideal throughput characteristic is a strictly linear function of the number of packet-handling modules without any upper bound with respect to the number of ISM's. Unfortunately, such a characteristic cannot be realized; the main sources causing throughput degradation follow.

First, we can distinguish among three different sources of overhead in the sense that if the processor utilization is not allowed to exceed a fixed threshold (for delay reasons) processor cycles get lost and, thus, packet throughput is reduced: (1) overhead due to communication between cooperating pairs of ISM's; (2) overhead due to communication between ISM and NCM; (3) overhead due to ISM storage overflow. If a request for packet buffer cannot be satisfied by a particular ISM, say ISM i, then this ISM will issue a new request and send it to its neighbor ISM, ISM i+1. If the request cannot be satisfied by ISM i+1 it will be passed to ISM i+2, etc. The overhead consists of actions like unsuccessful buffer search, issue new request for buffer, interpret request in next ISM, buffer search in next ISM, etc. Overheads 1 and 2 are of a linear type with respect to the number of modules, i.e., they only decrease the slope of the ideal throughput curve and do not constitute a major concern. Overhead 3 (due to overflow) is highly non-linear, an intuitive explanation being that if a request for buffer cannot be satisfied by a particular ISM the probability is very high that subsequent requests will also be unsuccessful. This means that in case of overflow a sizable amount of processor cycles can get lost.

Second, buffer overflow per se - apart from causing a reduction of useful processor cycles - can significantly reduce the number of packets handled per unit time: packets which cannot be accommodated in storage get either lost and have to be retransmitted later or the NCM has to activate a flow control mechanism which throttles further arrivals of packets. Again, because overflow occurs in clusters one has to expect that any overflow situation appreciably contributes to throughput degradation, i.e., a more than linear deviation from the ideal throughput curve. Two extreme situations are conceivable: either no overflow to a neighbor ISM is allowed or the whole chain of ISM's can be tested, if necessary. In reality a maximum number of ISM's will be specified to which requests for buffer can overflow, say three ISM's, because the requests have to be serviced in real time.

3.2 Main Features of the Simulation Model

Here we explain the main features of the simulation model to the extent necessary for the discussion of the results in the following section. Details are provided in the Appendix.

The simulation model cannot reflect all details of the real operation. Since ISM overflow seems to be the crucial point which determines the throughput characteristic, the model has been structured such that the major mechanisms and parameters on which overflow depends are modeled in adequate detail, whereas the intermodule communications contributing to overheads 1 and 2 are only modeled globally.

The most basic parameter which determines buffer overflow is the maximum amount of storage available in each ISM for storing of packets. This buffer storage is organized in pages where a page corresponds to the maximum packet size. In addition to the packet buffer area each ISM has to provide storage for programs and tables, where the table size grows with the number of ISM's (see Table 1).

Table 1

Total number of ISM's	2	4	10	20	40
48-kbyte memory					
Storage required for tables in kbytes	2	3.5	6.5	11.5	23.5
Number of pages per ISM for packets	40	34	22	2	-
64-kbyte memory					
Storage required for tables in kbytes	2.5	4	7	12	24
Number of pages per ISM for packets	102	96	84	64	16
96-kbyte memory					
Storage required for tables in kbytes	3.5	5	8	13	25
Number of pages per ISM for packets	226	220	208	188	140

The actual amount of buffer storage available at a particular point of time depends on several factors. First, on the buffer reservation strategy employed in boundary nodes due to the virtual-circuit concept (as opposed to datagrams). Note that transit nodes are unaware of this concept. Two extreme strategies have been implemented in the model: (1) One page is reserved per virtual circuit at connectio set-up time for the duration of a connection. If a message consists of more than one packet additional pages have to be obtained from a buffer pool as packets arrive. (2) No buffer is reserved and pages are fetched from a pool at packet arrival time. In this case the notion of connection and/or connection duration is irrelevant in our model. In transit nodes buffer always has to be acquired from a pool. The second factor which determines the actual amount of buffer storage available is the buffer hold time which in turn depends on the amount of processing per packet (instructions to be executed per packet), on the waiting times for processor, transmission and acknowledgments from adjacent nodes, on traffic load and speed of the outgoing lines.

Finally, the overflow mechanism per se has been modeled, whereas the ISM processors are not contained in the model, their utilization and, thus, their impact on throughput can be globally estimated from the simulation results.

4. NUMERICAL RESULTS

In this section we present and discuss results obtained from simulation runs. As has been pointed out before the question which is of prime interest is the throughput characteristic of the multi-ISM configuration operating under the control scheme outlined in Section 2 provided that the total traffic is of the packet-switched variety. This means we are primarily concerned with the throughput as a function of the number of ISM modules. Before discussing results we briefly summarize the underlying assumptions.

4.1 Assumptions

In order to obtain comparable situations we assumed that for a particular set of system and traffic parameters the work load originally offered to each ISM remains constant regardless of the number of ISM modules. To achieve this, the following assumptions have been made: (1) The total traffic offered to the switching node is a linear function of the number of ISM modules and is uniformly distributed among all modules. (2) Each ISM within a node has to handle — via an appropriate number of CAM's — the same number of subscriber lines and trunks. (3) All ISM's within a particular node are identical, i.e., contain the same number of pages available for storing of packets provided the number of ISM's is kept constant. The number of buffer pages depends on the total number of modules according to Table 1. (4) All subscriber lines and all trunks connected to the switching node have the same transmission speeds: 600 bit/sec and 9600 bit/sec, respectively, unless otherwise specified. (5) For all trunks a constant probability of 0.01 for retransmission of packets due to transmission errors and a constant delay of 0.05 sec for acknowledgments between adjacent nodes are assumed.

The distribution function of the packet length is given in the Appendix; for the results presented here we assumed a minimum packet length of 100 bits, maximum length of 2000 bits, and an average length of 300 bits.In order to obtain a broad view we investigated three types of switching nodes: two boundary cases where the switching node handles either local traffic only or transit traffic only and the realistic case where a certain percentage of the traffic is local and the remainder is transit traffic. In the following examples the locally generated traffic is 50% local (destined for other users of the same node); 50% long-distance (traffic going to other nodes). Both the long-distance traffic arriving at and destined for the node considered and the transit traffic equal the long-distance traffic generated locally.

4.2 Discussion of Results

The ideal throughput characteristic is a strictly linear function of the number of ISM modules; it serves in each of the diagrams as reference curve. It is obvious that the slope of the ideal throughput curve corresponds to the packet arrival rate per ISM.

We first study a typical throughput characteristic and the influence of the packet arrival rate (30, 40, 50 packets/sec) per ISM under the following assumptions, see Fig. 5. The node considered handles both local and transit traffic, each ISM has a 48-kbyte memory, requests for storage can overflow over three ISM's, and the buffer reservation strategy is such that no buffer is reserved and pages are fetched from a pool at packet arrival times (exponentially distributed interarrival times). As Fig. 5 shows, the real throughput initially follows the ideal throughput curve, then begins to deviate from it, and after reaching a maximum decreases. The number of ISM's where the deviation from the ideal curve begins and where the maximum is reached depends on the packet arrival rate. The explanation for this characteristic is as follows. The purpose of increasing the number of ISM's is to provide additional throughput capability if more terminals and trunk lines are to be attached or, in other words, more traffic has to be handled. As can be seen from Table 1 the number of pages per ISM which are available for storing packets decreases with a growing number of ISM's. The reason lies in the fact that with a growing number of terminals and lines the tables contained in each ISM also grow and, as a consequence, reduce the amount of buffer space for a given memory size. Fewer pages for storing packets, on the other hand, imply a higher probability of ISM buffer overflow and thus a reduction of throughput.

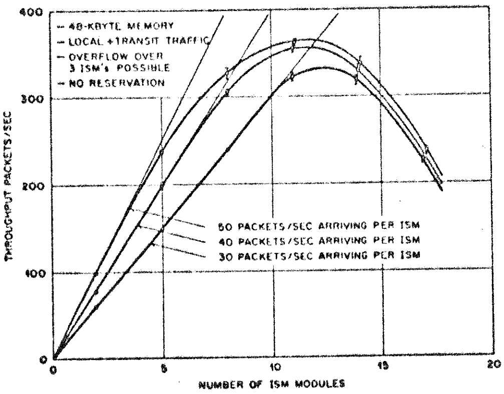

Fig. 5. Throughput versus number of ISM modules for different packet arrival rates

Figure 5 also contains the 95% confidence intervals. They have been measured for all subsequent curves as well, however are omitted in favor of clearer figures.

Next we analyze the impact of ISM memory size and of the overflow strategy, Fig. 6. Node type and buffer reservation strategy are the same as before, the packet arrival rate is 40 packets/sec. The figure clearly shows that with increasing memory size: (1) the maximum throughput is shifted to greater numbers of modules; and (2) the value of the maximum throughput significantly increases. In the case of a 48 kbyte memory the throughput reaches the maximum for 12 modules, whereas for a 64-kbyte memory 37 modules are possible before throughput degradation occurs. For an ISM memory size of 96 kbyte, the saturation point definitely lies beyond 40 modules. This indicates that by means of additional memory the maximum number of possible ISM modules is increased due to the reduction of overflow activity. Regarding the overflow strategy the following general observations can be made. If the case where requests for packet storage cannot overflow is considered as the reference case, then the greatest gain in throughput is obtained if one allows an overflow over three additional ISM's. Overflow over more than three ISM's does not appreciably contribute to a further increase of throughput but creates a timing problem.

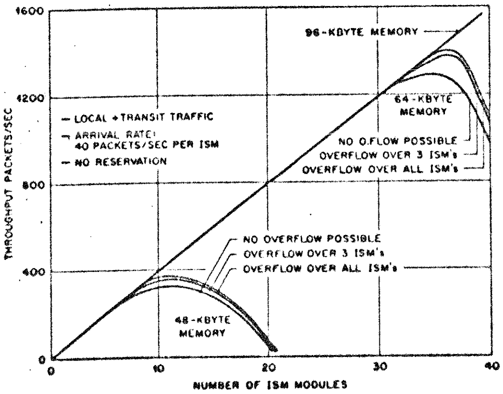

Fig. 6. Throughput versus number of ISM modules for different memory sizes and different numbers of possible overflows

In Fig. 7 we study the impact of the buffer reservation strategies employed in the boundary nodes as described in Section 3.2 under the following assumptions: node handling local and transit traffic, 64-kbyte memory, packet arrival rate 40 packets/sec, no overflow possible. In the case that one page (maximum packet size) is allocated at the time a virtual circuit gets established, we furthermore assume an average connection duration time of 60 sec and an average message length of 30 packets. Details regarding the distribution functions are contained in the Appendix. It can be seen that the reservation of buffer space on a virtual-circuit basis has a significant impact on system throughput compared with the strategy where no reservation is performed as in the previous figures. Other strategies, allocating only fractions of a full page to any active virtual connection or providing some extra buffer pool dedicated to all active connections, are expected to yield throughputs in the range delimited by the boundary cases shown in Fig. 7. Our simulation results indicate that it will — at least for moderate storage sizes — be only possible to reserve buffer space corresponding to a fraction of a full packet.

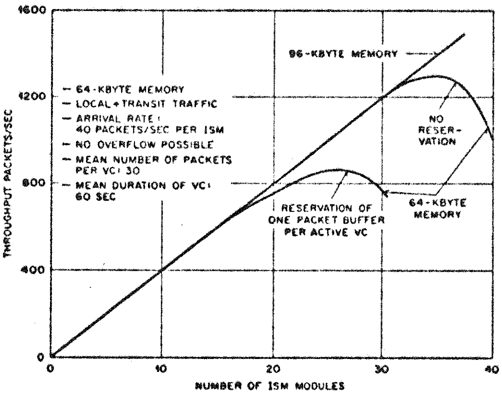

Fig. 7. Throughput versus number of ISM modules for different buffer reservation strategies

To further broaden the view we distinguish in Fig. 8 among three different types of switching nodes — handling local traffic only, transit traffic, local and transit traffic — as outlined in Section 4.1. The system and traffic parameters are: 64-kbyte memory, packet arrival rate 40 packets/sec, no ISM overflow, and no buffer reservation. The node which handles local traffic only reaches throughput saturation at approximately 1000 packets/sec with 30 ISM modules; the node being offered both traffic types can handle up to 1300 packets/sec with 36 ISM's. A node handling transit traffic only, however, does not show any saturation effect within the range considered. These differences are due to the different speeds of subscriber lines and trunks (600 bit/sec and 9600 bit/sec, respectively), which in turn significantly contribute to the buffer hold times.

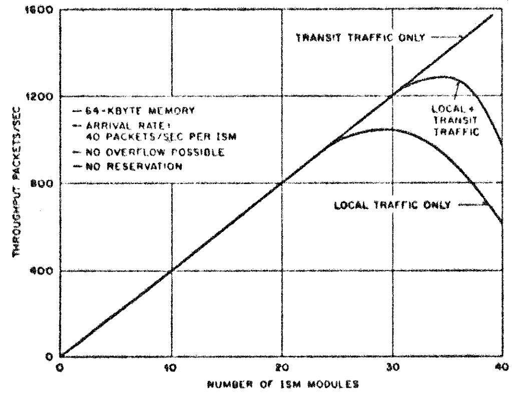

Fig. 8. Throughput versus number of ISM modules for different types of switching nodes

Using the same argument, i.e., decreasing buffer utilization with increasing transmission speed, the impact of the trunk speed on the throughput of a transit node, Fig. 9, can be explained.

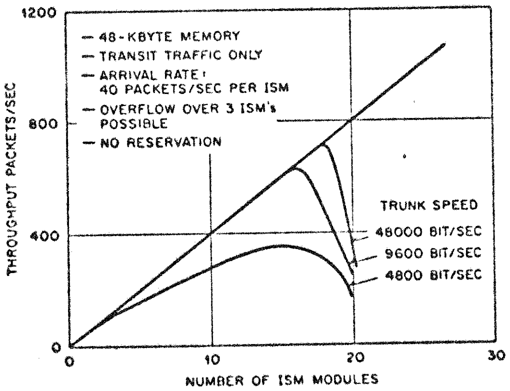

Fig. 9. Throughput versus number of ISM modules for different trunk speeds

Finally, Fig. 10 represents estimates of the ISM processor utilization for the most critical case of a node which handles transit traffic only. As mentioned in Section 3.2 the processors have not been explicitly modeled. Since the simulation yields the total number of requests to be serviced by the processors per unit time, we can estimate the processor utilizations under the assumption that no additional overflow of requests due to processor overload occurs. The following assumptions pertaining to processing have been made. ISM processor speed: 0.5 MIPS, instructions to be executed per packet in the data phase (including acknowledgments): 3000, instructions for a buffer request: 600 [1]. As long as the throughput behavior is ideal the processor utilization remains constant. In the nonlinear region the processor utilization follows the tendency of the throughput curve only if no overflow of requests is allowed. However, if overflow to other ISM's is possible, then the processor utilization grows rather rapidly due to the increasing overhead arising from unsuccessful buffer requests. Since the processor utilization should not exceed a certain threshold for delay reasons, the rapid increase of the utilization in case of heavy overflow activity further reduces throughput.

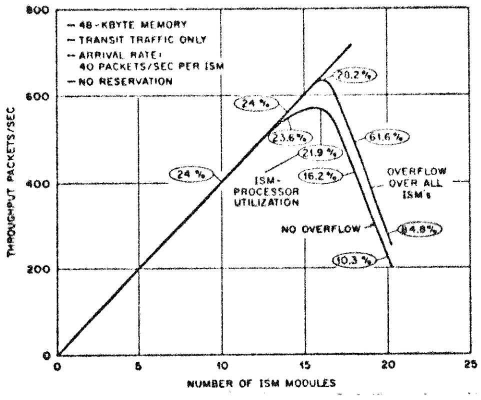

Fig. 10. Throughput versus number of ISM modules with corresponding processor utilizations

5. CONCLUSIONS

The major finding of this investigation is: the novel concept to achieve high throughput by reducing intermodule communications overhead allows the operation of the required number of processors without encountering saturation in the relevant range of throughput.

Nevertheless, the saturation effect does exist and limits the maximum number of permissible modules. Further findings in this regard are:

1) Storage overflow is the key effect which causes saturation. Hence, memory size of the packet-handling modules constitutes the prime parameter determining the throughput characteristic: with increasing memory size the maximum throughput is shifted to a greater number of modules and its absolute value significantly increases.

2) Overflow of requests for packet storage yields the greatest gain in throughput if overflow over three additional modules is allowed. Overflow over more modules does not appreciably contribute to a further increase of throughput but creates a timing problem.

3) Since availability of buffer resources determines the overflow activity, generally speaking, all parameters which contribute to the storage hold time of a packet have to be considered. In particular, it turned out that the strategy of how to allocate buffer space to newly created virtual circuits has a considerable impact on throughput, especially for small and medium memory sizes. Also, the speeds and utilizations of the outgoing lines show an impact on throughput which cannot be disregarded.

4) The results further show a significant increase of the processor utilization in case of heavy overflow activity which in turn can contribute to an additional throughput degradation provided this

utilization has to be kept below a certain threshold for delay reasons.

6. ACKNOWLEDGMENT

The authors would like to thank Prof. Dr. A. Lotze for his continued interest in the work, W. Schuster for writing the simulation program, and C.J. Jenny for helpful discussions.

APPENDIX

This Appendix gives an outline of the traffic model for the performance analysis. In a first section, the nature of a traffic model is briefly discussed together with a summary of those system aspects which are considered to be sensitive with respect to the system performance. In the second section, the global traffic model is developed describing the basic structure and the logical function. The global model is split into a number of specific submodels which will be described in greater detail in the third section.

1. Modeling Objectives

A traffic model describes the flow of "customers" (calls, packets, etc.) through a system. Herein, physical transmission paths or processing stages are modeled as "servers", whereas buffers for waiting customers are represented by "queues". Customer arrivals are generated by an "arrival process". The holding time of an occupied server is given by the "service process". Both arrival and service processes are usually described in terms of statistical distributions reflecting the random character of traffic. A set of operational modes by which resources are allocated to requesting customers (routing, scheduling, etc.) completes the definition of a traffic model.

The performance of the real system can be found by the analysis of a corresponding traffic model either analytically or by traffic simulation. Models for traffic simulation can generally be of a greater complexity allowing the description of rather arbitrary interrelationships among customers or between customers and the various resources, whereas analytically feasible models are confined to simpler interrelationships and random processes. The performance analysis for the packet-switching node described above has been based on a simulation model which includes a number of interrelationships between performance-sensitive quantities.

2. Global Traffic Model

The global traffic model is constructed from a number of specific submodels according to the logical structure of Fig. 11. The principal meanings of the various submodels are as follows:

Submodels	Function
VC ARRIVALS	Generation of interarrival times for virtual calls originating from those lines being associated to one ISM and resulting in the establishment of a virtual circuit.
PACKET ARRIVALS (VC)	Generation of number, interarrival times and lengths of packets offered during an established virtual connection.
PACKET ARRIVALS (TRUNK)	Generation of interarrival times and lengths of packets on an incoming trunk.
VC ROUTING	Determination of VC destination (local line/outgoing trunk).
PACKET ROUTING	Determination of packet type (destination/transit packet). Assignment of destination packets to VC's.
ISM BUFFERS	Description of buffer state (idle/reserved/occupied).
SUBSCRIBER LINES	Description of packet delay for waiting and transmission to local destination.
TRUNKS	Description of packet delay for waiting and transmission over one trunk to neighbor node.
ACKNOWLEDGMENT DELAY	Description of round-trip delay for acknowledgment signal (including two propagation delays, processing overhead, and transmission time in reverse direction).

The model consists of an arbitrary number M of ISM's each of them being associated to N_{TI} incoming trunks and an arbitrary number of subscriber lines. Each ISM has a buffer pool of S_{ISM} buffers for packets. There is a total of N_{TO} outgoing trunks and an arbitrary number of outgoing lines. Each outgoing subscriber line carries only those packets which belong to an associated established virtual circuit (VC), whereas trunks are shared by packets of many VC's.

Once a VC is generated, one buffer is reserved in the corresponding ISM. There is an arbitrary number of overflows possible ranging from 0 (no overflow) to M-1 (full accessibility of all buffers in the node). After occupation of the reserved buffer by an arriving packet of the particular VC, a new buffer is reserved. The reservation instant can be chosen arbitrarily between start and end of packet transmission. Similarly, for each incoming trunk one buffer is reserved dynamically as described previously.

The buffer occupation time consists of the arriving-packet transmission time, the waiting and transmission time for forwarding to

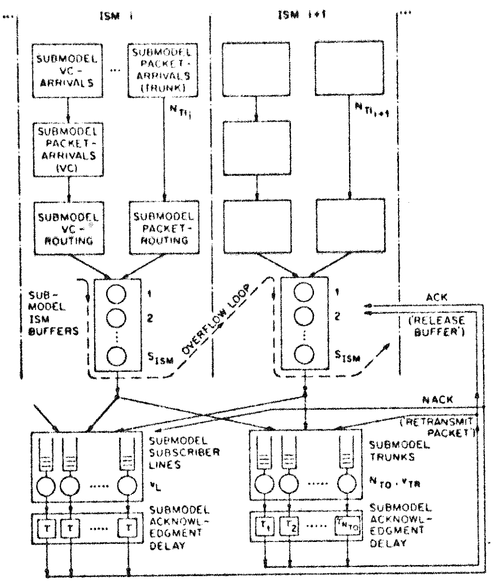

Fig. 11. Structure of the global traffic model

either the local destination or the neighboring node, the round-trip delay for acknowledgment and eventually those delays being incorporated by retransmissions in case of negative acknowledgments due to transmission errors.

Within an established VC, all packets belonging to this VC are routed along the same path. Path routing is accomplished at the setup of a VC. Arriving transit packets are treated independently; they are routed to the various outgoing trunks according to a probabilistic routing strategy.

3. Submodels

This section defines the major functions of the various submodels in greater detail.

3.1 Submodel VC ARRIVALS

Originating VC's are generated according to an arbitrary infinite source arrival process. The results presented here are based on exponentially distributed interarrival times $T_{VC,i}$ according to

$$P\{T_{VC,i} \leq t\} = 1 - \exp(- \lambda_{VC,i} t) , \qquad (1)$$

where $\lambda_{VC,1}$, the VC arrival rate of ISM_i, $i = 1,2,\ldots,M$.

3.2 Submodel PACKET ARRIVALS (VC)

For each established VC, the submodel generates: a) the number NP of packets offered for transmission; b) their interarrival time T_{AP}, and c) their length L.

The number of packets is generated according to an arbitrary discrete distribution:

$$P\{NP = j\} = p_j, \quad j = NP_{min}, \quad NP_{min} \; (\; 1, \ldots, NP_{max} \; . \tag{2}$$

The interarrival time T_{AP} between two successive packets within a VC may be arbitrarily distributed. The results presented are based on constant interarrival times according to

$$P\{T_{AP} \leq t\} = \begin{cases} 0, & t < a \\ 1, & t \geq a \end{cases} \; . \tag{3}$$

The packet length L defines — together with the line speed v_L — the packet transmission time. The distribution of L is according to

$$P\{L \leq x\} = \begin{cases} 0 & x < l_{min} \\ \left(\dfrac{x - l_{min}}{l_{max} - l_{min}} \right)^{\alpha} & l_{min} \leq x \leq l_{max} \\ 1 & x > l_{max} \end{cases} \; . \tag{4}$$

The parameter α is determined by the boundaries l_{min} and l_{max} and by the prescribed average packet length. Note that the special cases of a uniform distribution between these boundaries and of a constant distribution are included.

3.3 Submodel PACKET ARRIVALS (TRUNK)

Arriving packets from an incoming trunk are modeled through a single-server queueing system. The arrival process to this substitute system is assumed to be Markovian analogously to (1) with arrival rate λ_{TR}. This arrival process describes the arrival of packets at the *previous* switching node being routed to the node considered. The server stands for the incoming trunk.

3.4 Submodel VC ROUTING

The routing of packets is performed on a VC basis (path routing) at the instant of connection setup according to a probabilistic model. An established VC is local with probability p_{INT} or external with probability $(1-p_{INT})$. External VC's are routed to one of the N_{TO} outgoing trunks randomly. All packets belonging to the same VC are routed along the same path.

3.5 Submodel PACKET ROUTING

For each incoming trunk packet, the submodel defines: a) the packet type (transit/destination); b1) the outgoing trunk to where transit packets are routed, and b2) the VC to where destination packets are assigned. The determination of transit/destination follows through fixed proba-bilities p_{TRANS} and $(1-p_{TRANS})$, respectively. Transit packets are routed to the outgoing trunks randomly. Destination packets are assigned to a predefined number of simultaneously existing VC's randomly and will be routed accordingly.

3.6 Submodel ISM BUFFERS

The Intermediate Storage Modules (ISM) are modeled through their buffers numbered $1,2,\ldots,S_{ISM}$. Buffers are reserved for established VC's in boundary nodes and incoming trunks through a special buffer assignment routine. This routine also realizes the management of idle buffers (pool), the overflow loop, and the buffer release. The occupation time of buffers is defined through waiting, transmission and acknowledgment delay times.

3.7 Submodel LINES

The outgoing lines from the node to terminals or host computers are modeled as single-server queueing systems (one per established VC). The line transmission time is defined by the packet length L and line speed v_L.

3.8 Submodel TRUNKS

Outgoing trunks to the neighbor nodes are modeled as single-server queueing systems. A trunk transmits all packets routed to it according to a first-come/first-served schedule. The trunk transmission time is defined by the packet length L and trunk speed v_{TR}.

3.9 Submodel ACKNOWLEDGMENT DELAY

First, this model defines the round-trip delay of an acknowledgment including two propagation delays, processing overhead and transmission time in reverse direction. These delays are assumed to be constant, namely, τ_i, $i = 1,2,\ldots,N_{TO}$ for trunks and τ for local lines. Secondly, the model defines the acknowledgment signals ACK, NACK for buffer release or retransmission, respectively; this is accomplished through probabilities $p_{ACK,i}$, $(1-p_{ACK,i})$ for trunk $i = 1,2,\ldots,N_{TO}$ and p_{ACK}, $(1-p_{ACK})$ for lines, respectively.