April 79

N 5/4

# Throughput Considerations in a Multi-Processor Packet-Switching Node

W. BUX, P. KÜHN, AND K. KÜMMERLE

*Abstract*—The prime objective of the present paper is to analyze the throughput characteristics of a multi-processor configuration for packet switching—suggested in the context of an integrated network for circuit and packet switching—and to identify the major parameters on which it depends. 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 is shown that the proposed partitioning of the packet-handling function does indeed permit high throughput before saturation is encountered.

## 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 microprocessor), for reasons of: (a) cost; (b) ease in changing the packet-handling capacity upon change of load requirements, and (c) a very high degree of reliability.

Considering the high throughput requirement of about 1200 packets/second/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 packet-switching 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 and thus yields a throughput behavior which is expected to follow the ideal characteristic over the range required. The ideal throughput characteristic is a strictly linear function of the number of modules.

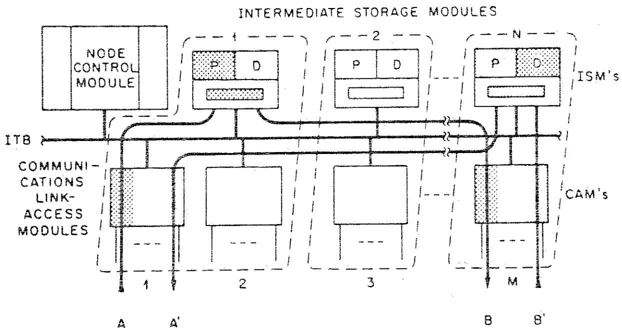
INTERMEDIATE STORAGE MODULES
NODE CONTROL MODULE
1
2
N
P
D
ISM's
ITB
COMMUNI-CATIONS LINK-ACCESS MODULES
CAM's
---
1
2
3
M
A
A'
B
B'

Figure 1. Modular structure of switching node.

The present paper is a more detailed investigation of the scheme proposed in [1], and a concise representation of the analysis contained in [3]. The prime objective is to analyze the throughput characteristic of the multi-processor 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. Although obtained for a specific system, the significance of the results lie 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 and explains the main features of the control concept; Section 3 discusses the parameters which determine the throughput characteristic, and contains a description of the simulation model; finally, Section 4 discusses results.

## 2. NODE OPERATION AND CONTROL MECHANISM

Figure 1 shows the configuration of the switching node. The following groups of modules are distinguished:

(a) the Node-Control Module (NCM) performs connection management and all node-control functions including error-control procedures;

(b) the Communication Access Modules (CAM) interface a variety of communication links;

(c) the Intermediate Storage Modules (ISM) provide buffer storage and processing capability for packet switching in the data phase, and finally,

(d) a bus interconnects all modules.

The objective of the packet-handling mechanism proposed in [1] is to reduce intermodule communications overhead to avoid the saturation effect typical of multiprocessor configurations. The basic idea 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.

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 $P$; see Fig. 1. The process *dispatch* is represented by the function $D$. It is responsible for managing the outbound traffic, a supervisory, not a processing function. $P$ and $D$ functions communicate by means of node internal messages.

Paper approved by the Editor for Computer Communication of the IEEE Communications Society for publication after presentation at EUROCOMP 78, London, England, May 1978. Manuscript received January 19, 1978; revised November 27, 1978.

W. Bux is with the Institute of Switching and Data Technics, University of Stuttgart, Stuttgart, Germany.

P. Kühn was with the Institute of Switching and Data Technics, University of Stuttgart, Stuttgart, Germany. He is now with the University of Siegen, Siegen, W. Germany.

K. Kümmerle is on leave at IBM Research Center, Yorktown Heights, NY 10598. He is with the IBM Zurich Research Laboratory, 8803 Ruschlikon, Switzerland.

0090-6778/79/0400-0745$00.75 © 1979 IEEE

A *D* function is created for each departing line or group of departing lines; dashed lines in Fig. 1. As far as the *P* function is concerned, CAM's know the addresses of ISM's assigned to perform the *P* function for each of their incoming lines. This assignment is preset. It is, however, not fixed, and may change according to traffic conditions. The basic operation is as follows. The *P* function has responsibility for a packet until the header is analyzed. Then the *P* function notifies the appropriate *D* function that a packet is ready for dispatching, whereby the packet is not physically moved. As soon as the *D* function realizes that the line is available it forwards a read request to the *P* function which then starts transmission of the packet. Upon reception of an acknowledgment, the *D* function notifies the *P* function which in turn releases the buffer. A special situation occurs when a chosen *P* function cannot perform packet handling due to there being no available storage: Upon arrival of a packet the CAM generates a request for buffer and passes it to the ISM where the assigned *P* function resides. If the request cannot be granted, it is forwarded to the neighboring ISM and so on until an ISM is found which can fulfill the request. We denote this operation of forwarding requests to the next ISM in case of buffer depletion as *overflow* mechanism. A detailed discussion and further explanations are provided in [1].

## 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. There exist two main effects causing throughput degradation: Overhead which leads to a reduction of useful processor cycles and blocking of the incoming traffic if no buffer is available.

The exchange of data between source and destination is based on a virtual-circuit (VC) concept: at first, a logical path is set up between source and destination nodes; during the subsequent data phase, all packets belonging to that VC are routed along the established path. Within both source and destination nodes, packet buffers can be reserved for every active VC. The reserved buffer space is returned to the common buffer pool at VC take-down.

*Reduction of Processor Cycles:* Three different sources of overhead cause a reduction of useful processor cycles: (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. The latter 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. Overhead (3) is highly non-linear.

*Blocking of Traffic:* The nonavailability of buffer can significantly reduce the number of packets handled per unit time: packets which cannot be accommodated in storage, i.e., in any ISM, 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. Since the number of packets which cannot be handled during a blocking situation shows a clustering effect, it is to be expected that any blocking situation appreciably contributes to throughput degradation, i.e., a more than linear deviation from the ideal throughput curve. Both categories of sources causing throughput degradation will be quantitatively studied. In addition, the throughput gain due to the overflow mechanism will be analyzed by comparing two extreme strategies: 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.

TABLE 1
SIZE OF PACKET STORAGE AS A FUNCTION OF NUMBER OF ISM's

| 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 most basic parameter determining buffer overflow is the maximum amount of storage available in each ISM for packet storing. 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: (1) with the number of ISM's and (2), slightly, with the ISM storage size; Table 1. Each ISM contains five major tables: a page table, reflecting the state of every buffer page and, if allocated, to which local line or trunk it is assigned. The size of this table depends on the ISM storage size. The other tables, source table, routing table, associated ISM table, and destination table, contain entries pertaining to connections which have been established, line status, routing, accounting, and the assignment of ISM's to outgoing lines and trunks. These tables grow with the number of ISM's. Further details are to be found in [4].

### *3.2 Features of the Simulation Model*

Here we explain the features of the simulation model whose structure is shown in Fig. 2. Since ISM overflow seems to be the crucial point determining the throughput characteristic, the model has been structured such that the major mechanisms and parameters on which overflow depends are modeled in detail. Current technology allows bus architectures with sufficient bandwidth for handling the traffic loads projected for this particular environment; therefore the bus mechanism is not included. Although the delays being caused by processing are neglected, the ISM processor utilization, however, can be estimated from the frequency of processing requests generated in the simulation program.

The model consists of an arbitrary number *M* of ISM's, each of which is associated to an arbitrary number of incoming trunks and subscriber lines. Each ISM has a buffer pool of $S_{\text{ISM}}$

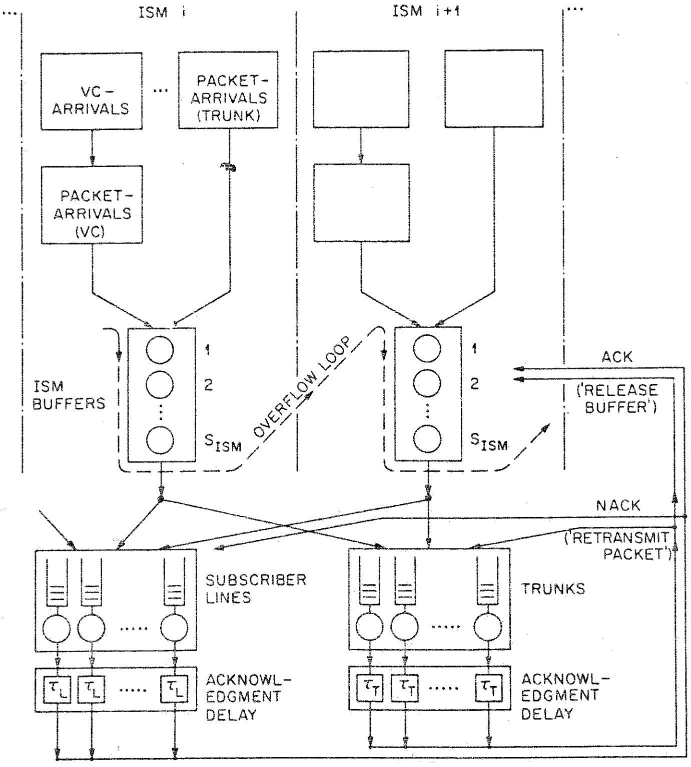

Figure 2. Structure of the simulation model.

buffers for packets. The outgoing trunks as well as the outgoing lines are modeled by single server queues with first-come first-served discipline.

Furthermore, the VC's can be operated either with or without buffer reservation. In case of VC's with buffer reservation, one buffer is reserved in the corresponding ISM's in the source and destination nodes for every active VC. The maximum number of ISM's to which a buffer request may overflow can be prescribed between 0 (no overflow) and $M - 1$ (full accessiility of all buffers in the node). After occupation of the reserved buffer by an arriving packet of the particular VC, a new buffer in the same ISM is reserved. Similarly, for each incoming trunk one buffer in the corresponding ISM is reserved dynamically. The buffer occupation time consists of the arriving-packet transmission time, the waiting and transmission time for forwarding to either the local destination or the neighboring node, the round-trip delay for acknowledgment and possibly those delays being incorporated by retransmissions in case of negative acknowledgments due to transmission errors.

VC's are generated according to an arbitrary infinite source arrival process. The results presented subsequently are based on the assumption that the time intervals between the generation of successive VC's are exponentially distributed. For each established VC, the simulation generates: the number of packets offered for transmission, their interarrival time, and their length. The number of packets is generated according to an arbitrary discrete distribution. The interarrival time between two successive packets within a VC may be arbitrarily distributed. The results presented are based on constant interarrival times.

The packet length $L$, finally, defines—together with the transmission speeds—the packet transmission times over lines and trunks. It can be arbitrarily prescribed; for the results shown we chose a distribution of $L$ with upper and lower boundaries $L_{\min}$ and $L_{\max}$, respectively, according to

$$P\{L \leqslant x\} = \begin{cases} 0 & x < L_{\min} \\ \left(\dfrac{x - L_{\min}}{L_{\max} - L_{\min}}\right)^{\alpha} & L_{\min} \leqslant x \leqslant L_{\max}. \\ 1 & x > L_{\max} \end{cases}$$

The parameter $\alpha$ 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.

Packets arriving from an incoming trunk are modeled through a single-server queueing system. The arrival process to this substitute system describes the arrival of packets at the *previous* switching node which are forwarded to the node considered. This arrival process is assumed to be Markovian; the server stands for the incoming trunk.

The routing of packets is performed on a VC basis at the instant of connection setup according to a probabilistic model. All packets belonging to an established VC are routed along the path. Arriving *transit* packets are treated independently; they are randomly routed to the various outgoing trunks.

## 4. NUMERICAL RESULTS

In this section we present and discuss results obtained from simulation runs. As pointed out previously, the question of prime interest is the throughput characteristic of the multi-ISM configuration operating under the control scheme outlined in Section 2. 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 were 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 bits/s or 9600 bits/s, respectively. (5) For all trunks, a constant probability of 0.01 for packet retransmission due to transmission errors and a delay of 0.05 s for acknowledgments between adjacent nodes are assumed.

The distribution function of the packet length is given in Section 3.2; 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. As far as the traffic is concerned, we distinguish between local and transit traffic. The local traffic leaves the switching node on a (low-speed) subscriber line, whereas the transit traffic is carried on a (high-speed) outgoing trunk line. Both are either generated locally

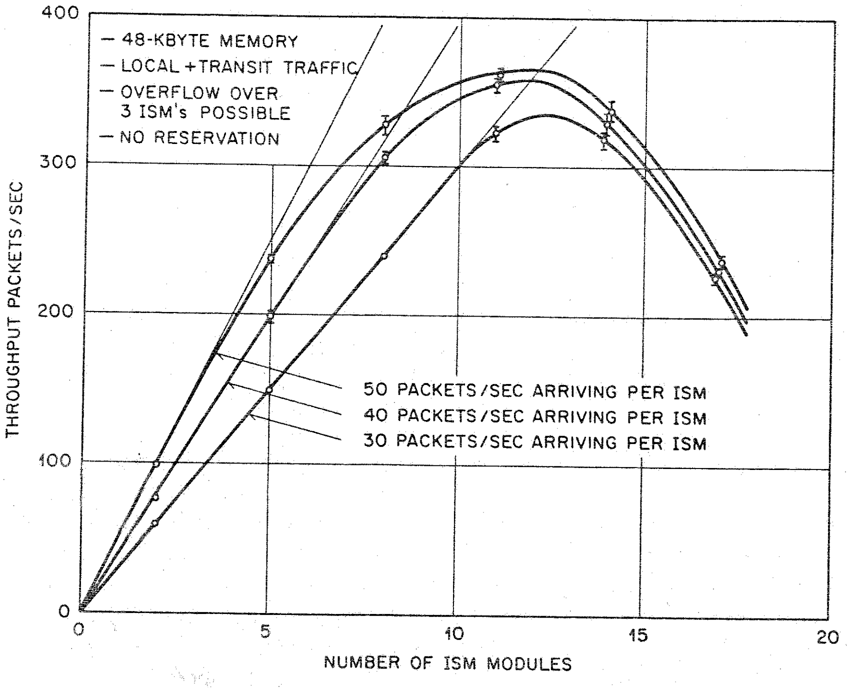

Figure 3. Throughput versus number of ISM modules for different packet arrival rates.

or arrive from an adjacent node. In the following examples the locally generated traffic is 50% local and 50% long-distance. 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 a 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/s) per ISM under the following assumptions; see Fig. 3. 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. As Fig. 3 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 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 as discussed in Section 3.1. Fewer pages for storing packets imply a higher probability of ISM buffer overflow and thus a reduction of throughput.

Figure 3 contains the 95% confidence intervals. They were also determined for all subsequent curves; however, they are omitted in favor of clearer figures.

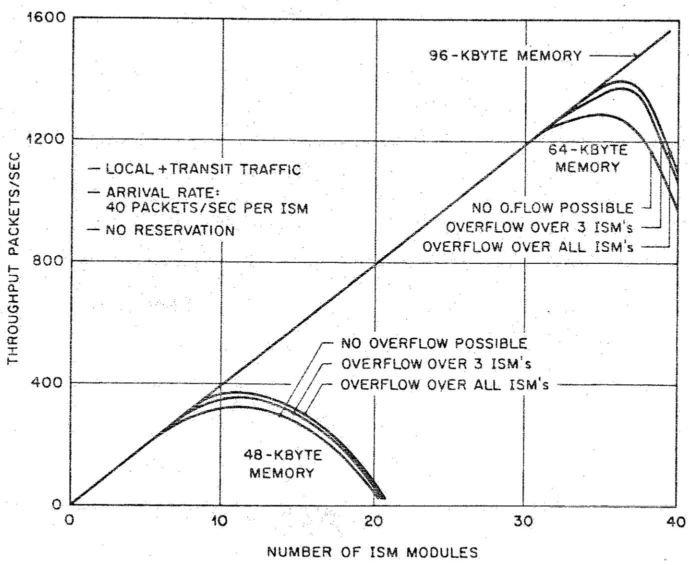

Figure 4. Throughput versus number of ISM modules for different memory sizes and different numbers of possible overflows.

Next we analyze the impact of ISM memory size and of the overflow strategy; Fig. 4. Node type and buffer reservation strategy are the same as before; the packet arrival rate is 40 packets/s. 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

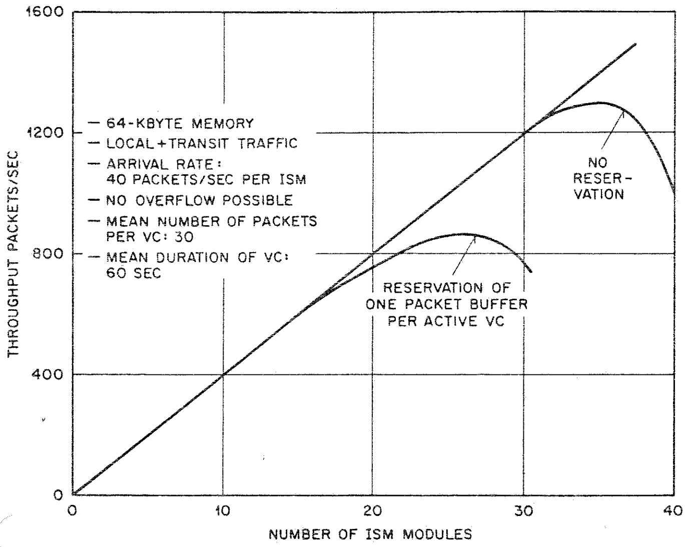

Figure 5. Throughput versus number of ISM modules for different buffer reservation strategies.

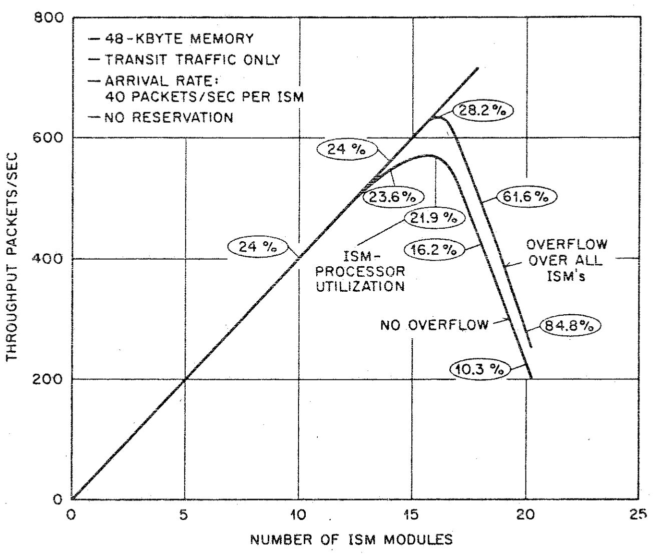

Figure 6. Throughput versus number of ISM modules with corresponding processor utilizations.

due to the reduction of overflow activity. Regarding the overflow strategy the following general observations can be made. The nodal throughput obtained with the packet-handling mechanism suggested is adequate even if no request overflow for packet storage is allowed in case of storage saturation in a particular ISM. The throughput, however, can be further increased by providing the overflow mechanism as discussed in Section 2. Figure 4 shows that allowing an overflow over three ISM's yields the greatest additional throughput. Overflow over more than three ISM's does not appreciably contribute to a further increase of throughput but creates a timing problem.

In Fig. 5 we study the impact of the buffer reservation strategies employed in the boundary nodes due to the virtual-circuit concept. Two strategies have been implemented in the model: (1) One page is reserved per virtual circuit at connection 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.

The following assumptions were made: node handling local and transit traffic, 64-kbyte memory, packet arrival rate 40 packets/s, no overflow possible. In the case that one page (maximum packet size) is allocated at the time a virtual circuit gets established, we assume furthermore an average connection duration time of 60 s and an average message length of 30 packets. 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 two cases shown in Fig. 5. Our simulation results indicate that—at least for moderate storage sizes—it will only be possible to reserve buffer space corresponding to a fraction of a full packet.

Finally, Fig. 6 represents estimates of the ISM processor utilization for the most critical case of a node which handles transit traffic only. 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 were 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 request overflow 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.

## 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 causing 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 an additional gain in throughput which is most pronounced 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.

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.

## 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, C. J. Jenny for helpful discussions, H. R. Rudin for reviewing the manuscript, and the reviewers for constructive suggestions.

## REFERENCES

[1] C. J. Jenny, K. Kümmerle, "Distributed processing within an integrated circuit/packet-switching node," *IEEE Trans. on Commun.*, Vol. COM-24, No. 10, pp. 1089-1100, October 1976.

[2] "European Computer and Communications Markets 1973-1985", PA International Management Consultants and Quantum Science Corp., London, England and New York, 1973.

[3] W. Bux, P. Kühn, K. Kümmerle, "Distributed processing in a packet-switching node: performance analysis," in *Proceedings of the European Computer Congress,* 1978, London, England, pp. 165-185.

[4] C. J. Jenny, K. Kümmerle, H. Bürge, "Network node with integrated circuit/packet-switching capabilities," IBM Research Report RZ 720, August 1975.