Discussion Record

SESSION NO. 61 - S.P.C. SWITCHING SYSTEMS 1

Chairman - H. INOSE (Japan)
Discussion Leader - G. DIETRICH (Germany)

G. DIETRICH: Session 61 covered three of seven papers dealing with "S.P.C. Switching Systems". The invited paper presented by Mr Villar gave us an excellent overview about the broad range of traffic studies in the field of SPC systems, including even definitions of traffic terms like nominal, overload and maximum call handling capacity. It would be very good to have commonly accepted, well defined terms not only for us, the traffic engineers, but even more for people applying our results and operating SPC switching systems. I hope, CCITT will be soon successful as Mr Chapuis promised yesterday.

Another remarkable point is that during the early design phase of SPC systems, traffic support is very essential, to an extent much higher than for the previous system generation. We have to give advice knowing only very few eye numbers of access case has been treated in the paper of Gerrand and Gueraro.

The paper of Weisschu and Wizgall was a typical example for a call-type simulation, applying the classification of simulation techniques proposed. Furthermore, their paper has shown that analytical methods have a good chance to be successful in the SPC control area. More examples for this are given in the next session. Thank you.

Paper No. 611
Author: J.E. VILLAR (Spain)

O.A. PEDERSEN (for R.H. LAUFFENBERGER) (U.S.A.): Has not the author in oversimplifying the mechanics of function or load sharing systems ignored a critical consideration? In order to provide any reliability or fault detection in either of the duplexed load sharing processors, it is my understanding that the two independent/interdependent processors communicate on every call attempt. Thus in addition to the fixed overhead there is a call dependent overhead in the non-involved processor while the involved one is actively processing a call. It would seem that in periods of overload this could require severe derating if grade of service is to be maintained. I believe the French P.T.T. has documented this conclusion. Would the author care to comment on this? Also, what calculation procedures does this necessitate when the duplexed units approach maximum processor occupancy?

J.E. VILLAR: While I have no doubt, provided a very simplified description of the control mechanisms in an SPC system. I have been careful to use the word overhead in its strict traffic engineering sense of traffic independent processing, and not in its loose software sense. The processing to which you refer occurring in one processor as a result of an interprocessor message from another processor, is always included in the category of traffic dependent processing, when making one's calculation.

It is important to realize that in overload conditions, number of interprocessor messages per call does not increase. Their impact is the same as other traffic dependent tasks during overload conditions.

As a consequence, in a load sharing system, if we apply the criteria that the GOS for both nominal and overload conditions are met by the complete set of processors, and that when a single processor fails during nominal load traffic, the system should be capable of meeting the normal GOS criteria, there is not any derating. Furthermore, the system will have a considerable spare capacity in duplex operation, that can be utilized in overload.

For instance, our simulation results using a call type model, show that the MC duplex SPC will handle a 20% increase in Erlangs together with a 60% increase in BHCA over normal rated offered, without introducing any derating.

You say that you believe the French PTT has published a document on time that I have no knowledge of any such document, and I have asked some of the French representatives at this Congress, and they have no knowledge of the document. If you can give me further details, I will follow up the matter.

In answer to your final question, in the extreme overload situation in which both processors approach or reach their maximum occupancies, we do not know a practical method of analyzing the system capacity, except the use of a full call simulation.

Our full call simulation models have served us well in comparing the effectiveness of different overload control mechanisms.

Paper No. 612
Authors: H. WEISSCHUH and M. WIZGALL (Germany)

P. GERRAND (Australia): Concerning the collection of input data for your analysis and simulation, I was interested to read in your paper that you performed direct measurements on your system, using a "time monitor", in order to estimate program run times. Could you explain to us what your time monitor is, and how it works; and how you treat program run times that vary considerably, such as path-search times?

H. WEISSCHUH:

1. In order to measure the program run times of the switching programs one must generate the real input messages for the common control. This was done by an environment simulator which had been developed for program debugging. The system software of our common control contains a facility module, called "time monitor" for run time measurements. This facility module traces the times when a switching program is started and when it has finished. The traced values are stored on a magnetic disk. From these the run times can be evaluated by a special evaluation program. We have determined the mean program run times as well as their distribution functions. The mean values as well as the coefficient of variation are shown in table 1 of our paper.

2. For our call type simulation we have approximated the measured distribution functions for the run times by Erlangian-k pdf. The orders of k of the Erlangian-k pdf are also given in table 1.
T. SAITO (Japan): I appreciate your paper as an interesting model of PCM switching system. I wonder is there some difference between PCM systems and space division systems from the viewpoint of the traffic behaviour of SPC systems. Could you comment on the difference?

M. WIZGALL: To answer your question, let us look at Figure 7 of our paper. The difference between PCM switching systems and space division switching systems is given by two features:

- the transmission principle
- the switching principle.

Concerning the question, only the switching network and the incoming and outgoing speech paths have to be considered. In both cases the incoming and outgoing speech paths can be treated in the same way, that means from the viewpoint of the common control exists no difference between PCM channels and analog trunks.

The switching network has to be treated in different ways for PCM and space division switching, respectively. Because the traffic behaviour of a SPC switching system is determined by

- the switching network and
- the common control

the traffic behaviour of the whole systems is naturally influenced by the traffic behaviour of the switching network, which depends on the type of the switching network (digital or analog).

E.K. CHEW (Australia): My questions refer to your results in diagram 1, 2 and 4.

1. In these diagrams you have assumed zero overheads. I wonder if non-zero overheads, say 10 - 20% of a sampling period, would affect your results significantly. Did you consider this?

2. I also wonder if you have investigated the relationship between Ycpu and RTF (ie the speed of the CPU).

Could you please comment? Thank you.

M. WIZGALL: To your first question:

In diagrams 1, 2 and 4 we have assumed the overhead equal to zero.

Of course we have performed simulation runs with overhead. The overhead for these runs was chosen to 10% of the sampling interval. At each sampling instant a constant overhead phase is performed.

Therefore the total load of the CPU is the sum of the load of the CPU due to the switching programs and the load due to the overhead. eg the total load of the CPU will increase to 70% if the load due to the switching programs is 50% and the overhead is 20%. The effect of the overhead eg on the waiting times of messages in front of the CPU are shown in Figure 5 of our paper.

2. We have introduced the RTF to study the influence of the computing speed on the system performance. It is assumed that the corresponding times of the switching programs are linearly increased by means of RTF.

Therefore the shape of the curve for the load on the CPU is the same for all RTF's.

The influence of the RTF on the waiting times can be seen in Figure 5 of our paper.

G. DIETRICH: I have difficulties to understand your diagram 7. What is the reason for the tremendous long queues for case 1 and 2 at the beginning of the simulation, which are going down to zero later on?

M. WIZGALL: Diagram 7 of our paper shows the actual queue length of the queue in front of the common control vs. time. In order to explain the shape of the curves, let us look only on curve 1 which is also shown in the diagram below.

This diagram shows in the upper part the arrival rate, in the centre part the number of occupied PCM channels and in the lower part the queue length.

At time 0 the system is empty. Just after time 0 the arrival rate of call attempts is increased as a step function from \( \lambda = 0 \) to \( \lambda = 35 \), which is approximately 10 times the normal value.

Due to the fact, that the system is empty at time 0 most of the call attempts are accepted. Therefore the number of busy channels increases rapidly as shown in the diagram. This results in a very large number of requests for the common control which cannot be served immediately.

Therefore the queue length increases.

At the point, where nearly all PCM channels are busy the automatic overload protection mechanism in the concentrator control becomes effective and limits the number of accepted call attempts by rejecting call attempts. The queue length decreases approximately at the point where nearly all PCM channels are occupied. This is due to the fact, that for rejected calls no requests for the common control are generated. Therefore the number of arriving requests for the control becomes lower than the number of served requests. This leads to the decrease of the queue length.

After about 60 sec the steady state queue length for heavy overload is reached. Then, there are about 1 to 10 waiting requests in the queue (hatched in the diagram). The queue length stagnates in this range because the overload protection mechanism remains active.
G. DIETRICH: Concerning the analytical part of your paper, is your message that SPC systems can be solved analytically? The presented analytical and simulation results compare remarkably good.

H. WEISSCHUH: The model which was investigated analytically is a submodel, extracted from the total switching system. The input process is a cued batch input. This process can be described by the batch size probabilities. We determine these values in the simulation program and describe therefore very accurately the actual input process for the calculation. The distribution function of the service times was also measured in the simulation program for a prescribed call mix. It fitted very well to a negative exponential distribution function. Because of the accurate description both of the input and the service process, we have obtained good results. Generally spoken, queuing analysis can be applied to such submodels where the input process as well as the service process and queuing strategies can be described close to reality. For complex models, e.g., with sampled feedbacks, there exists today no exact analytic solution. Therefore, up to now, simulation techniques are the only tool to analyze such systems. Furthermore, if we want to investigate the transient behavior of such complex systems, we can do this only by simulation techniques.

Relating to your question, one can state that detailed models for SPC switching systems cannot be solved exactly up to now. However, submodels can be treated by queuing analysis. This fact has also been pointed out in the invited paper by Mr. Villar.

For traffic engineering approximative methods today exist for complex queuing networks which apply the method of decomposition. An example can be found in the paper of Dr. Kühn. This method can be used for complex network structures without sampling systems.

R.C. ADDIE (Australia): Since an evenly distributed overhead may be considered to slow down a processor by a certain rate there should be a correspondence between the introduction of an overhead of, say 10%, and a corresponding change of the RTF factor in your paper. Could you comment upon this please?

M. WIZGALL: Concerning eg the load of the CPU, there is a correspondence between the overhead and the RTF, if the overhead is related to the load of the CPU. In the investigated system, at each sampling instant a constant overhead phase occurs. Therefore the load of the CPU is increased by a constant amount. The variable overhead defined in our paper is not considered in the presented investigations. This overhead may correspond to the RTF. But, concerning eg the waiting times of messages in front of the common control, there will be no correspondence between overhead and RTF.

A.E. JOEL (Jr) (U.S.A.): At ISS '74 I presented a paper attempting to define the advantages of fully stored program control (FSPC) systems. Your paper proposes a less than FSPC system where decisions are made in the periphery without leaving a record in the central processing unit. From a traffic point of view why should not all queues be located within the central processor?

M. WIZGALL: First a comment on the statement that our switching system represents a "less than FSPC system". The control concept of our switching system bases on a central control with preprocessing units. The preprocessing units perform some simple switching functions and diminish in that way the load of the common control. But for all accepted calls being in progress, records are maintained in the common memory of the central control. The processing in the central control is carried out by reference to instructions stored in a common memory. The preprocessing units are microprogrammed controlled. Therefore all switching functions are performed by instructions stored in a memory, i.e., our system represents a full stored program control system.

Concerning your question, I would like to point out: The transfer of information between the central processor and the preprocessing units is performed in fixed time intervals (clock period t, c.f. Figure 5 of the paper). Messages from the peripheral preprocessing units to the central processor are buffered in a queue until the sampling instant. At the sampling instant the messages are transferred into the input queue of the central processor. The buffering in the peripheral units allows to process more than one event during one sampling period. This leads to an effective utilization of the preprocessing units.

Paper No. 613
Authors: P. GUERRAND (Australia) and A. GUERRERO (Spain)

G. DIETRICH: Your paper is a good example for traffic support during system design, which I consider an important task in traffic engineering.

My question is, can your solution for the periodic polling be extended to other than constant holding times? This is of interest because sometimes a marker has to work consecutively several times (1 hr, 2 hr, 3 hr, ... ) to complete a job.

On page 4 you refer to Figure 10, but there is no Figure 10.

P. GUERRAND: Firstly, thank you for drawing our attention to a misprint in our paper: the reference to Figure 10 on page 4 should have been to Figure 8. In reply to your main question, our analytical solution for periodic polling is strictly only applicable to constant holding times. However, the method may be applied to the case you mentioned, in which a marker performs a sequence of different jobs each with a fixed holding time, for the same call by convolving the waiting time distributions for the individual jobs.