A SIMULATION STUDY OF COST OF DELAYS IN COMPUTER SYSTEMS

S.R. Clark
Department of Mathematics
University of Victoria
Victoria, B.C., Canada

and

T.A. Rourke
Department of Computer Science
University of Manchester
Manchester, England†

SUMMARY

A simulation model of a hypothetical time-sharing system with a finite non-contiguous store and an infinite auxiliary store is used to study the effects upon the mean cost of delay to all jobs processed by varying CPU utility, supervisor overheads, store size, etc.

It is shown that when the mean cost of delay per job is added to the mean system cost per job well formed minima appear indicating an optimum CPU utility, store size and CPU speed for the balance of these two costs.

Comparisons are made between the results of the simulation model and previous analytical work. In addition a further comparison is made between the simulation model and a real system. All comparisons were found to be very favourable.

INTRODUCTION

The purpose of this paper is to demonstrate the effectiveness of a partial cost analysis of computer system performance by simulation methods. The competing factors of cost of delay to jobs and the mean system cost averaged over each job are considered in the analysis. Job characteristics are generated from a set of input means by Monte Carlo methods, and the processing of these jobs is simulated using a next-event type of model incorporating the scheduling logic and the gross hardware characteristics of the computer system.

The results of such a partial cost analysis should be combined with other cost considerations to gain an overall cost picture. Rather than replacing accepted measures of system performance such as benchmark tests, the analysis may rely on these methods to provide some of the input information, such as the mean amount of central processor time requested. The characteristic feature of the type of analysis described here is the inclusion of the cost of delay to jobs as a relevant cost consideration.

The cost of delay to jobs has been considered by Greenberger[1] and Rasch[2] using analytical methods. Analytical methods, however, become unsuitable when non-linear cost curves (curves representing cost of delay to jobs as a function of delay) are used[1], or when space-sharing systems* of finite size are considered.

Scherr's simulation analysis of the Project MAC system at MIT and comparison of the results with actual statistics[3] has done much to validate the assumption that events which occur on a time scale of less than about 1 msec. can be ignored in a simulation of job throughput.

The analysis carried out here involves a simple hypothetical computer system which closely resembles the analytical system studied by Rasch[2]. Based upon the success of Scherr's simulation, events such as memory cycles which occur on a time scale of less than 1 msec. are ignored.

BRIEF DESCRIPTION OF THE MODEL

The simulation model used simulates the queuing system shown in Figure 1. The simulated current time is stepped forward to the time of the next event each time the next event is determined. The possible events are the arrival of a new job into auxiliary store, the start, continuation or termination of a time-slice, and the termination of a job.

The simulated current time is advanced by the supervisor cycle time, t_s, after the determination of the next event. Thus, in order to transfer control from one job to another two increments of t_s occur: one before the CPU time of the job being stopped is updated, and one before the next job is selected for processing. Whenever the execution of a time-slice of a job is interrupted by the arrival of a new job into auxiliary store the job being processed stays at the head of the execution list.

The waiting queue is ordered in a first-come-first-served manner within each priority class, the low priority jobs (the larger ones on the average) being towards the end of the queue. Whenever there is space available in main store, the first job in the waiting queue which can be accommodated is selected. The transfer time from auxiliary to main store is ignored.

It is important to stress that many of the simplifying assumptions made in this analysis could be very easily altered to suit a real environment.

The job stream consists of 3 priority classes, each of which has a different cost curve representing the way in which cost increases with delay. The arrival times, amount of CPU time requested, the size of the job in arbitrary units of store, and the priority class are supplied by the job generator using a series of input means and pseudo random number generators. The interarrival times of jobs in

*'Space-sharing' is used here to represent systems in which more than one job shares the main memory simultaneously. This includes paging systems.

†Now at: Technical support unit, Ministry of Technology, London.

each priority class are exponentially distributed and the distribution of CPU times closely resembles that observed for the University of Michigan system[4]. The distribution of job sizes closely resembles that observed for the Atlas system[5].

The highest priority class (class 1) consists of short jobs which arrive frequently, and the lower priority classes (classes 2 and 3) consist of longer jobs which arrive less frequently.

A number of the input parameters to the model are varied to study the effects on cost of delay. These parameters are therefore of direct interest. The variations are made relative to a standard setting, the results for which appear in each variation. The parameters are:

ρ - the CPU utility ($0 < \rho < 1$); the value of ρ for the standard calculation is 0.75 corresponding to a mean interarrival time of jobs, R, of 3.4 secs. ρ is varied indirectly by varying the mean interarrival time of jobs, leaving the mean CPU time request for the standard system at 2.6 secs.)

t_{rr} - the desired round-robin cycle time; (when one round-robin cycle is completed the time t_{rr} is divided among the jobs in the execution list to give time slices for the next cycle. This produces a cycle oriented discipline. The standard setting of t_{rr} is 6 secs.)

t_s - the supervisor cycle time; (the standard setting of t_s is 0.01 secs.)

W - the size of the main store in units; (the mean storage request is about 5 units and the standard setting of W is 50 units.)

v - the speed of the CPU. (the standard setting of v is unity). The values of the standard settings are either representative of real systems or were chosen to be realistic.

Figure 2 shows the cost curves used for each priority class in the calculation of the cost of delay to a job. These curves are an extension of the intuitive idea of a general cost curve given by Greenberger[1]. The abscissa represents the elapsed time multiplication factor[6] (ETMF) of the finished job, which is simply a ratio of the delay experienced from start to finish of a job to the time requested by the job. The mean cost of delay was taken as

$$C = \sum_{i=1}^{n} C_i/n \qquad (1)$$

where n is the number of jobs processed. In each calculation of the following analysis 1500 jobs were simulated.

The mean system cost per job was calculated using very simple assumptions concerning the cost of the components of the system. It is assumed that the system cost of the standard system is equally divided among the 3 components: CPU, auxiliary storage and main storage. The variation in the costs of system components is assumed to be linearly related to size or speed and to be continuous (for any real system it would of course be discrete). The system cost per job for the standard job stream is taken to be 0.2 cost units.

RESULTS

The five input parameters to the simulation model described above were each varied in turn, all parameters other than the one being varied having the standard setting. Figures 3 to 7 inclusive show how the mean cost of delay varies with the appropriate parameter. Where the mean system cost is not a constant this is shown too, and the addition of the two curves gives the "total cost."

In the 3 cases where the total cost is shown, a well-formed minimum is apparent. Such minima are likely to occur in real environments whenever the magnitudes of the mean cost of delay and mean system costs are similar. The minimum in the mean cost of delay in Figure 4 is very similar to the ones observed by Greenberger[1] and Rasch[2] analytically, using linear and exponential cost-curves respectively.

In addition to the mean cost of delay curve obtained by varying t_s with all other parameters in their standard setting (the curve marked 'time-slicing' in Figure 5), a second series of runs were carried out differing only in that the t_{rr} was set at a very large value causing a process-to-completion scheduling of the CPU. The cost advantages of time-slicing are apparent when t_s is less than 0.04 secs. but for larger values process-to-completion is less costly. This figure is in close agreement with the equivalent time of 0.035 secs. determined by Rasch analytically.

VALIDITY OF THE RESULTS

This simulation study is intermediate between the complexity of a real system and the complexity which can be solved analytically[1,2]. It was therefore of interest to see how the model would handle calculations analogous to previous analytical work,* and it is important to compare the simulation results with the results of a real system to ensure that the extension of the analysis to a real environment is in fact possible.

The simulation model described earlier was extended to include a finite sized magnetic drum and magnetic tape drives in order to carry out a simulation of the Atlas paging computer system for which there are published statistics[5]. Six weeks logging information gathered from Atlas in 1968 was used to obtain additional statistics, broken down into priority classes.

Figure 8 shows that the simulation results compare favourably with the Atlas statistics, demonstrating that the partial cost analysis would be effective in a real environment.

This successful comparison with the Atlas statistics gives further support to the comparison made by Scherr[3]. The distinction between the two comparisons is that Atlas is a paging system which allows the overlapping of CPU activity with tape and

*The results from the simulation model compared very favourably with Greenberger's analytical equation[1] and Kleinrock's analytical equation[7].

drum transfers, and in which the sizes of the main store and the magnetic drum are of importance. In the simulation carried out by Scherr it was not necessary to include the size of the main memory and CPU activity was not overlapped with drum/disk transfers.

SIMULATION TIMES

The simulation model is written in FORTRAN IV and was run on the IBM 360/65 at the University of Manitoba. The core size for the execution phase is 45K bytes. The choice of language was made in conformity with Nielsen's work[8] on the simulation of the IBM 360/67 and in conformity with Scherr's avoidance of standard simulation languages[3]. The prime objective in this choice was speed, but still retaining the power of a high-level language.

The time taken for a run of the model not including paging activity is about 2 minutes for the simulation of 1000 jobs. To some extent this figure is dependent upon the parameter settings and the resultant queuing. With little queuing the figure is smaller; for much queuing, it is larger.

It was with alarm that the time for a simulation with paging was noted. Atlas executes a page transfer for every 0.03 seconds of CPU time on the average, and this gives rise to very small differences between event times in the model. Figure 9 shows how the run time varied with the onset of paging ($k = 0$ indicates no paging; $k = 1$ indicates paging at the Atlas rate).

CONCLUSIONS

The partial cost analysis of computer system performance involving the mean cost of delay could be extended relatively easily to any non-paging system. Paging systems take considerably more computer time to simulate, however, and more caution is required. (A possible solution is to split the model into two sections, one to simulate millisecond activity and a second to simulate in a large time scale.)

When the mean cost of delay is comparable in size to the mean system cost well-formed minima appear in the "total cost" curve as the interarrival time of jobs, the size of the main store and the speed of the CPU vary. The implication of these minima is that the "total cost" should be a relevant consideration in the process of choosing a computer system or modifying an existing system in order to achieve better service.

In circumstances where the mean system cost does not vary, the mean cost of delay can give useful information concerning the optimum round-robin cycle time or the comparison of scheduling policies.

At this point it would be interesting to apply the analysis in a real environment where the mean cost of delay is as important as the mean system cost.

ACKNOWLEDGEMENTS

We wish to thank Dr. J. Blatny, Mr. B.W. McDonald, Mr. W.P. Boote and Mr. G.Riding for their help, and to acknowledge the financial assistance of the National Research Council of Canada. The simulation runs were carried out at the University of Manitoba.

REFERENCES

1. Greenberger, M. The priority problem and computer time-sharing. Management Science 12 (July 1966), 888-906.

2. Rasch, P.J. A queueing theory study of round-robin scheduling of time-shared computer systems. J. ACM 17 (January 1970), 131-145.

3. Scherr, A.L. An analysis of time-shared computer systems. Research Monograph No. 36, M.I.T. Press (1967), Cambridge, Mass.

4. Arden, B. and Boettner, D. Measurement and performance of a multiprogramming system. Second ACM Symposium on Operating Systems Principles, Princeton University 1969, 130-140.

5. Morris, D., Summer, F.H., and Wyld, M.T. An appraisal of the Atlas Supervisor. Proc. ACM Natl. Conf., Vol. 22 (August, 1967), 67-75.

6. Stimler, S. Some criteria for time-sharing system performance, Comm. ACM 12 (January 1969), 47-53.

7. Kleinrock, L. Time-shared systems, a theoretical treatment. J. ACM 14 (April 1967), 242-261.

8. Nielsen, N.R. Computer simulation of computer system performance. Proc. ACM Natl. Conf., Vol. 22 (August 1967), 581-590.

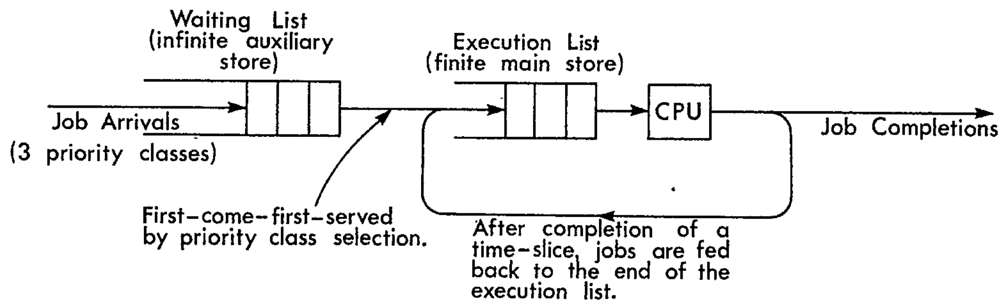

Fig. 1 Flow of jobs through the model.

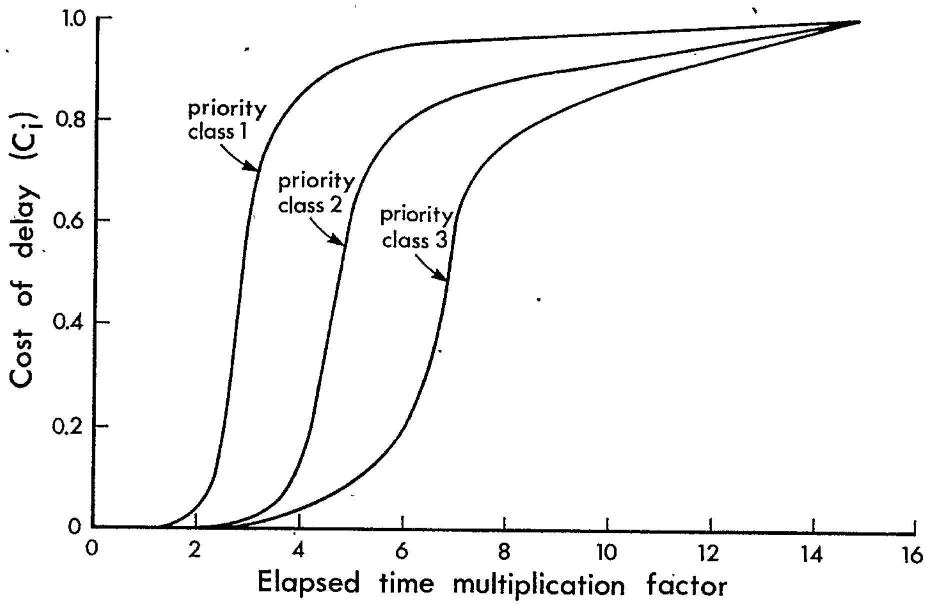

Fig. 2 Cost curves used to calculate cost of delay.

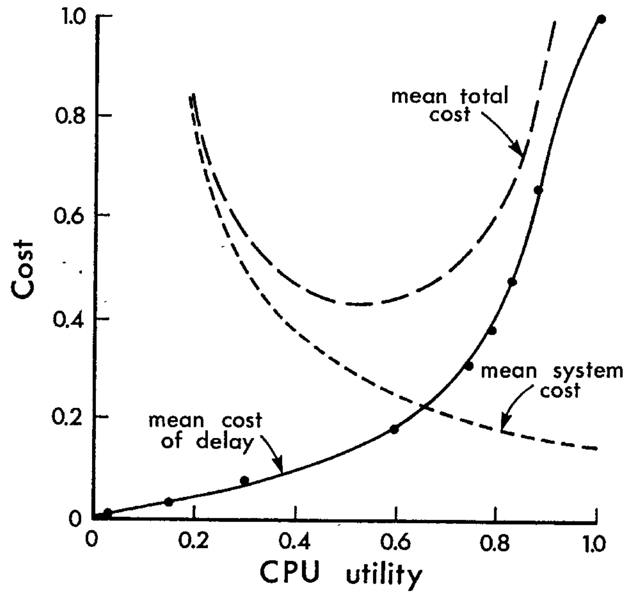

Fig. 3 Variation of cost with CPU utility.

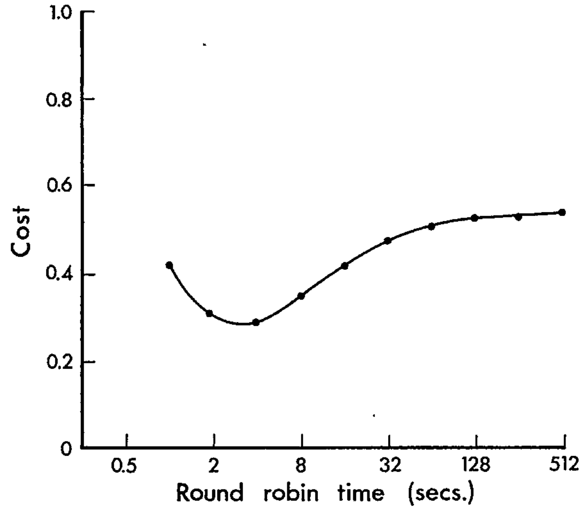

Fig. 4 Variation of cost with round robin time.

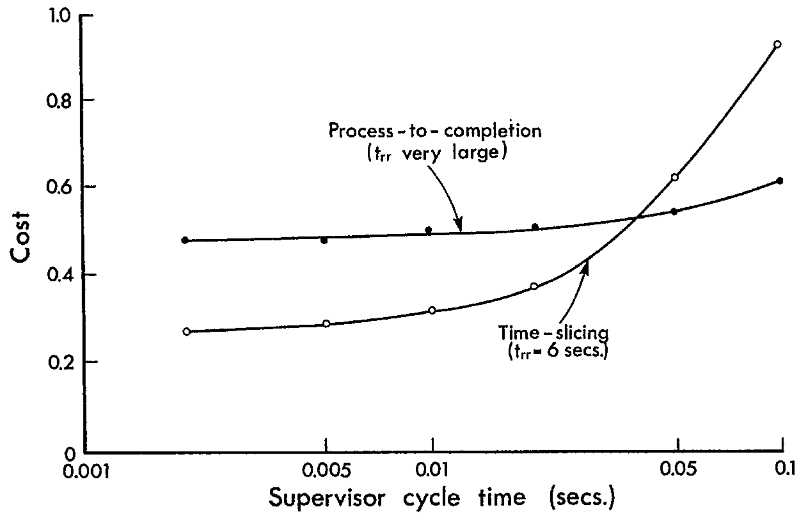

Fig. 5 Variation of cost with supervisor cycle time.

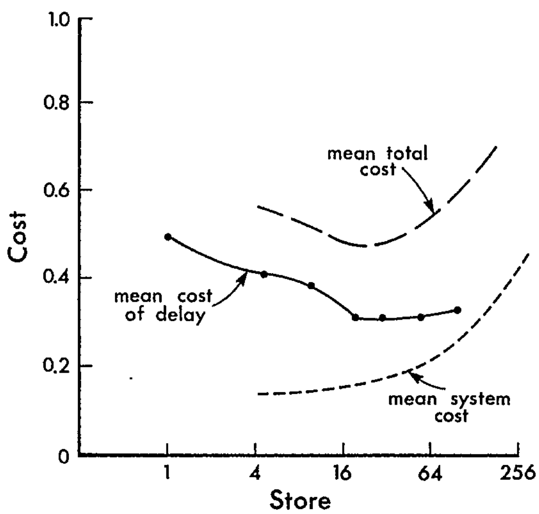

Fig. 6 Variation of cost with size of store.

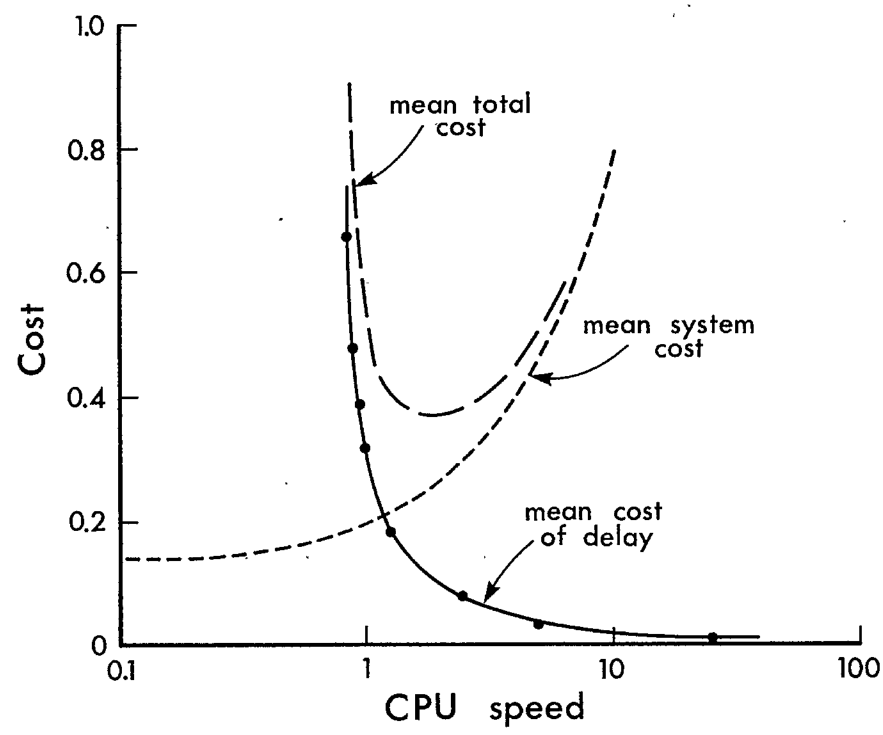

Fig. 7 Variation of cost with the speed of the CPU.

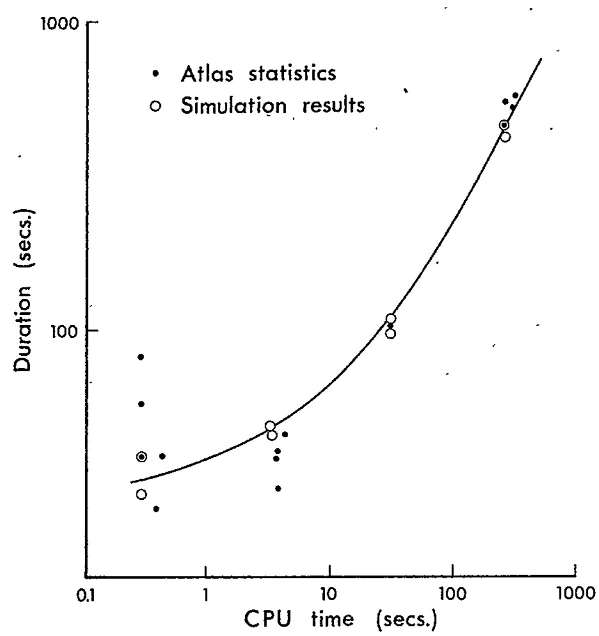

Fig. 8 Comparison between the Atlas statistics and the simulation results.

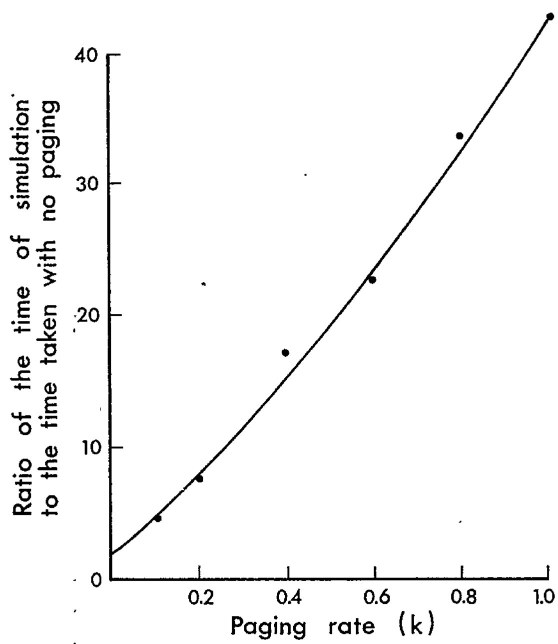

Fig. 9 Variation of simulation time ratio with the rate of paging.