MODELLING
A Tool for the Design and Optimization of Computer Systems

by

E. Gelenbe

October, 1976 TR-63

Paper consists of lectures delivered to the Department of Computer Sciences, The University of Texas at Austin, during October, 1976.

Department of Computer Sciences
The University of Texas at Austin

Modelling: A Tool for the Design and Optimization of Computer Systems

Part I. Random Injection Control of Multiprogramming in Virtual Memory

Part II. Stability and Control of Packet Switching Broadcast Channels

Part III. A Non-Markovian Diffusion Model and Its Application to the Approximation of Queueing System Behavior

E. Gelenbe

Universite de Liege and IRIA-LABORIA
Liege, Belgium France

Lectures delivered to the Department of Computer Sciences, The University of Texas at Austin, during October, 1976.

RANDOM INJECTION CONTROL
OF MULTIPROGRAMMING IN VIRTUAL MEMORY

E. Gelenbe, A. Kurinckx

We propose a new method for the control of a multiprogrammed virtual memory computer system. A mathematical model solved by decomposition permits to justify that the method avoids thrashing. Simulation experiments are used to test the robustness of the predictions of the mathematical model when certain simplifying assumptions are relaxed and when a slightly simpler control technique based on the same principle is used. Comparisons are given with the case where an "optimal" control is used and with that with no control.

Nous proposons une nouvelle méthode de contrôle de la multiprogrammation d'un système à mémoire virtuelle. Un modèle mathématique d'un système géré par cette méthode est résolu à l'aide de la méthode de décomposition. Des expériences de simulation permettent de vérifier les prédictions du modèle et de comparer la méthode proposée avec le contrôle "optimal" et avec le comportement d'un système sans contrôle.

I. INTRODUCTION

It is well-known that virtual memory computer systems exhibit an inherently unstable behaviour if processes are allowed into the common memory with no control : [2] is an early analysis of the subject and [3] provides a theoretical explanation. This instability translates itself in practice by a system throughput which becomes close to zero and very high turnaround or response times. The analysis in [3] provides an indication of how system control to avoid this effect can be used : it suffices to limit the multiprogramming degree N (defined as the number of processes allowed to share main memory) to a value below N' such that $\Theta(N')$, the system throughput for this value of N, is still at an acceptable level, where $\Theta(N)$ is predicted to have the form shown on Figure 1(b). Some of the results in [3] have been summarized in [4].

Using a mathematical and simulation model [6] developped to predict the throughput $\Theta(N)$ of a virtual memory paged computer sys-

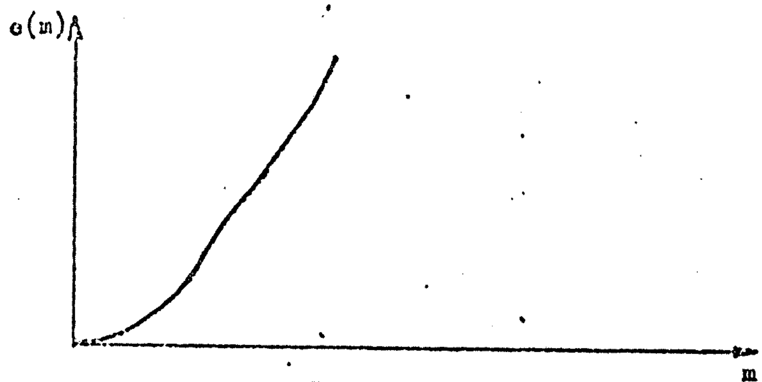


Figure 1(a)

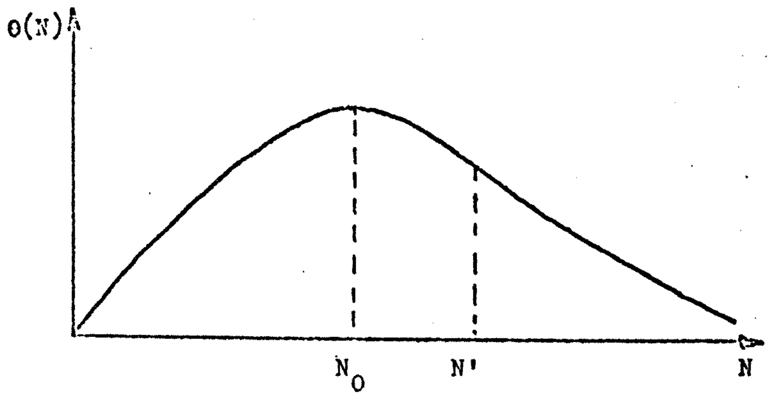


Figure 1(b)

tem, the problem of its feedback regulation so as to (i) avoid thrashing, and (ii) optimize its performance had been examined in [7]. Since the parameters of active processes such as paging rates for a given memory allocation, input-output rates, total CPU execution times, are necessarily time-varying, the approach used was an adaptive optimization algorithm. In view of the fact that characteristic of Figure 1(b) will vary with time, the algorithm in [7] to maintain the degree of multiprogramming close to its optimal value N_0 is a statistical maximum searching technique. This approach leads to a relatively elaborate estimation procedure.

A comprehensive review of the issues of virtual memory performance and control is given in [8], and [9,10] contain suggestions and evaluations of heuristic control policies.

In this paper we investigate a new technique for virtual memory system management which calls for less information gathering than that of [7] : it does not necessitate an instantaneous estimate of N_0, the optimal degree of multiprogramming but only an estimate of the throughput. An alleviation of supervisor functions and a greater possibility for decentralized management are potential advantages. This method avoids thrashing and operates the system close to optimum; it is not "optimal" however in the sense of the policy developed in [7].

In Section 2 a description of the method we propose and an approximate analysis are given. Section 3 is devoted to the presentation of simulation experiments showing the effectiveness of our approach and its comparison with the optimal control.

1.1. Random injection control.

Consider the idealized queueing network representation of the virtual memory system of Figure 2(a). A set of terminals

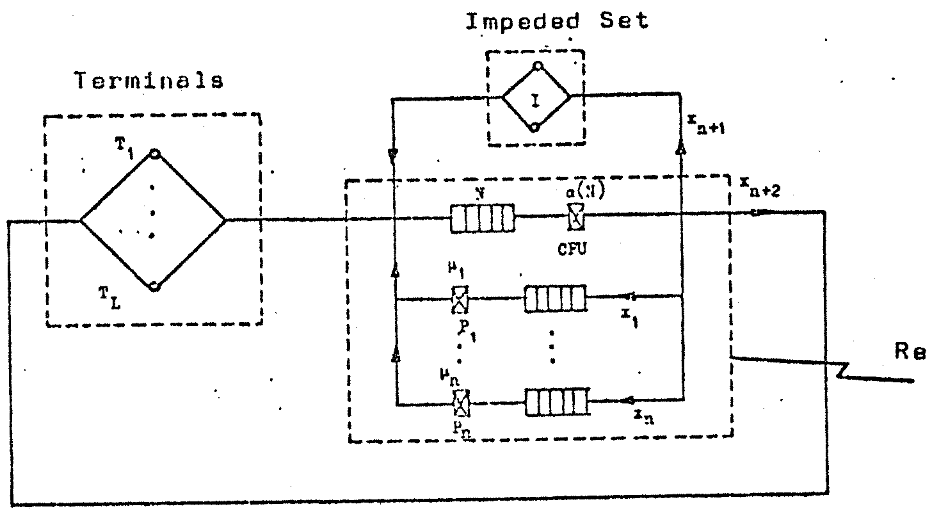


Figure 2(a)

$T_1,\dots,T_L$ generate processes and these are immediately accepted into the CPU queue. Processes which are either in the CPU queue or in that of one or more of the peripheral or secondary memory unit queues $P_1,\dots,P_n$ are allowed to acquire main memory. Assuming a demand-paging operation, main memory space will be acquired by an incoming process in an alternating sequence of page faults and CPU executions. When it has acquired a total of Q units of CPU execution time the process is placed in the set I of impeded processes and liberates the main memory space it occupied. For each impeded process a time interval is drawn at the end of which the process will be injected back into the active set, i.e. into the set of processes which share memory and other system resources. This time interval of duration τ will be chosen as a function of system parameters, and more particularly as a function of the throughput rate of the set of useful resources of the system (the "resource loop" of Figure 2(a)). Thus, in a way, the <u>impeded set of processes can be viewed as being in an artificial "think" state which is analogous to the "think" state of processes at terminals</u>. Though this is not necessary, we shall assume that τ is a random variable so that our control scheme is called <u>random injection control</u>.

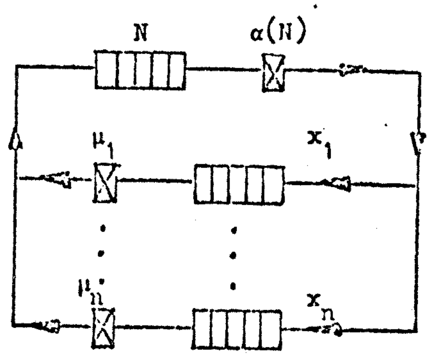


figure 2(b)
Resource loop

2. DESCRIPTION AND ANALYSIS OF THE PROPOSED CONTROL POLICY

The aim of this section is to present and justify the random injection control method for virtual memory computer systems.

We will first present a mathematical model which is used as a framework for justifying and evaluating the approach; this model is not new [6] and has already been used in [7] to evaluate an adaptive control policy for optimizing a virtual memory system. Then, the random injection policy will be described. It will be analyzed in two parts. We first use approximate analysis using the decomposition method [3,5] to show how random injection control avoids thrashing. Then simulations will be used to confirm these results and to show that it yields system performance comparable to that of an "optimal" control policy.

2.1. The mathematical model

In the system of Figure 2(a) a total of L processes are either in think state at one of the terminals, or they are in impeded state (at I in Figure 2(a))or in the CPU queue or at one of the I/O devices $P_1,\ldots,P_n$. We shall model the system under the following assumptions :

- the time spent in think state by a process before entering or re-entering the CPU queue is exponentially distributed of parameter λ .
- service at P_i is allocated on a first-come-first served basis ; consecutive service times are i.i.d. (independent and identically distributed) random variables, exponentially distributed of parameter μ_i.
- as in [6,7] the relationship between allocated primary memory space m and page-faulting behaviour of a process is established via the Belady-Kuehner life-time function e(m) [1] shown in Figure 1(a) and the following model. In process time, the consecutive inter page-fault intervals of a process are i.i.d. exponential random variables of parameter $[e(m)]^{-1}$.

Similarly the consecutive I/O inter-request intervals to device P_i($2 \le i \le n$, P_1 being the paging drum) is exponentially distributed of expectation a_i. Furthermore the total CPU execution time of a process between two terminal interactions is exponential of expectation C, and the total CPU execution time between two consecutive epochs it spends in the impeded state is exponential of expectation Q. Under these assumptions it is easily shown [6] that an uninterrupted execution interval of a process at the CPU is exponentially distributed of parameter
$\alpha = [(e(m))^{-1} + \Sigma_2^n(a_i)^{-1} + C^{-1} + Q^{-1}]$. At the end of such an epoch, it can be shown [6] that the process directs itself with probability :

- $x_1 = (\alpha e(m))^{-1}$, to P_1
- $x_i = (\alpha a_i)^{-1}$, to P_i , $2 \le i \le n$
- $x_{n+1} = (\alpha Q)^{-1}$, into the impeded set
- $x_{n+2} = (\alpha C)^{-1}$, to its terminal

- total primary memory space is R
- the time τ spent in impeded state by a process is exponentially distributed of parameter δ .

Remark 1 : if the terminals and the impeded set were non-existent in

Figure 2, and if N is the total number of processes in the system circulating among the CPU and the peripheral devices, then the curve of Figure 1(b) is representative of the CPU utilization at statistical equilibrium as a function of N.

Remark 2 : as stated in Section 1.1, the value of N (the degree of multiprogramming) is the sum of the number of processes which are either in the CPU queue or in the peripheral device queues $P_1,\ldots,P_n$. Thus the processes in think or impeded states do not share memory. Also, in our mathematical model, all processes are statistically identical and share main memory equally; thus m = R/N.

Remark 3 : in the simulation results given in Section 3 the service times at the peripheral devices are not exponential, and neither is the total CPU execution time of a process between two successive visits to the impeded state. Comparison between theoretical and simulation results will illustrate the robustness of the exponential assumptions (see also [6,7] concerning this point).

- for the life-time function e(m), we will use the usual form $e(m) = dm^k$ where d and k are constants. This particular form does not affect the theoretical analysis but simply the numerical results obtained.

2.2. The random injection policy

The random injection policy for virtual memory management operates on the following principles. Some of these principles have already been given as assumptions of the model in previous sections; we repeat them here for clarity.

P1 - Only the processes which are either in the CPU queue or in the queues of devices $P_1,\ldots,P_n$ share memory. As soon as a process enters the impeded set or leaves the CPU queue in order to return to the think state at a terminal, its main memory pages become available for other processes.

P2 - A process entering the impeded state will remain there for a random time duration, function of N, the degree of multiprogramming and of (M-N), the number of processes in impeded state ; M is the total number of processes which are not in think state. At the end of this time the process joins the CPU queue. This random time is exponentially distributed of parameter $\hat{\delta}(N,t)$ which is time dependent. In our policy we set

$$\hat{\delta}(N,t) = \begin{cases} B \text{ constant} , & N = 0 \\ h\hat{\beta}(N,t) , & N \quad 1 \end{cases} \tag{1}$$

where h is a constant to be determined below, and $\hat{\beta}(N,t)\Delta t$ is the probability that in $(t,t + \Delta t)$ a process leaves the resource loop to enter either the impeded or the think state. The constant B in (1) will be taken to be very large ($B = \infty$, for all practical purposes) since when $N = 0$ there is no reason to retain a process in the impeded set (i.e. its entry and exit are simultaneous).

2.3. Analysis of the random injection policy

In this section we shall present the equations governing the mathematical model of the virtual memory system controlled by the random injection policy. Only the equilibrium equations will be solved. It will be noticed that the exact solution to these equations is not known. Therefore an approximate solution based on the decomposition method of COURTOIS[3,5] will be presented.

2.3.1. The equations for the model

Consider the stochastic process $(V_t^i)_{t\epsilon R^+}$ representing the number of processes at the CPU queue if $i = 0$, or at the queue of P_i if $1 \leq i \leq n$, or in impeded state if $i = n + 1$, and in think state (at the terminals) if $i = n + 2$. We are interested in the joint probability distribution, for $t \geq 0$,

$$p(v,t) = Pr(V_t^0 = v_1, \ldots, V_t^{n+2} = v_{n+2} | V_0^0, \ldots, V_0^{n+2})$$

where $v = (v_0, \ldots, v_{n+2})$ is a vector of integers $0 \leq v_i \leq L$, $i = 1, \ldots, n+2$ (L is the total number of terminals and of user processes in the model). Under our assumptions the $p(v,t)$ satisfy the following system of differential difference equations :

$$\frac{d}{dt}p(v,t) = -[(L-M)\lambda + \Sigma_1^n \mu_i + \alpha(N) + D(v)]p(v,t) \tag{2}$$
$$+ (L-M+1)\lambda p(c(v,0,n+2),t)$$

$$+ \Sigma_1^n \mu_i p(c(v,0,i),t)$$

$$+ \Sigma_1^{n+2} (a_i)^{-1} p(c(v,i,0),t)$$

$$+ D(c(v,0,n+1))p(c(v,0,n+1),t)$$

where $a_{n+1} = Q$, $a_{n+2} = C$, $a_1 = e(m)$, and for any vector v we define

$$c(v,i,j) = (v_0,\ldots,v_i-1,\ldots,v_j+1,\ldots,v_{n+2}), \forall i,j,j \neq i \quad (3)$$

and in (2) any $p(\hat{v},t)$ such that $\hat{v}$ contains an element which is negative or larger than L is set to $p(\hat{v},t) = 0$ for all $t \geq 0$. Also we take $D(v) = 0$ if the (n+1)-th component of v is zero (i.e. $v_{n+1} = 0$). These conditions, together with the assumption that $\mu_i = 0$ in (2) if $v_i = 0$, suffice to define the boundary conditions of (2). Finally, notice that :

$$L = \Sigma_0^{n+2} v_i \ , \ M = \Sigma_0^{n+1} v_i \ , \ N = \Sigma_0^n v_i$$

D(v) is the arrival rate of processes from the impeded set to the CPU queue :

$$D(v) = (M-N)\hat{\delta}(N,t) \quad (4)$$

This system of differential difference equations will posess a unique solution $p(v,t) > 0$ such that $\Sigma_v p(v,t) = 1$, $t \geq 0$, if all the state transition rates are positive and finite. However the available theory [11] does not provide the equilibrium solution. In the following section we resort to an approximate solution method.

2.3.2. Approximate solution via decomposition

The approximate analysis technique for queueing networks developed in [3,5] is suited to systems composed of subsystems where the time constants of inter-subsystem interactions are appreciably larger than those associated with interaction within each subsystem. This makes it possible to use equilibrium results for each subsystem in the global system model.

In our analysis we apply the technique several times to our model. These successive simplifications are shown on Figures 3(a) and 3(b), for the model of Figure 2(a).

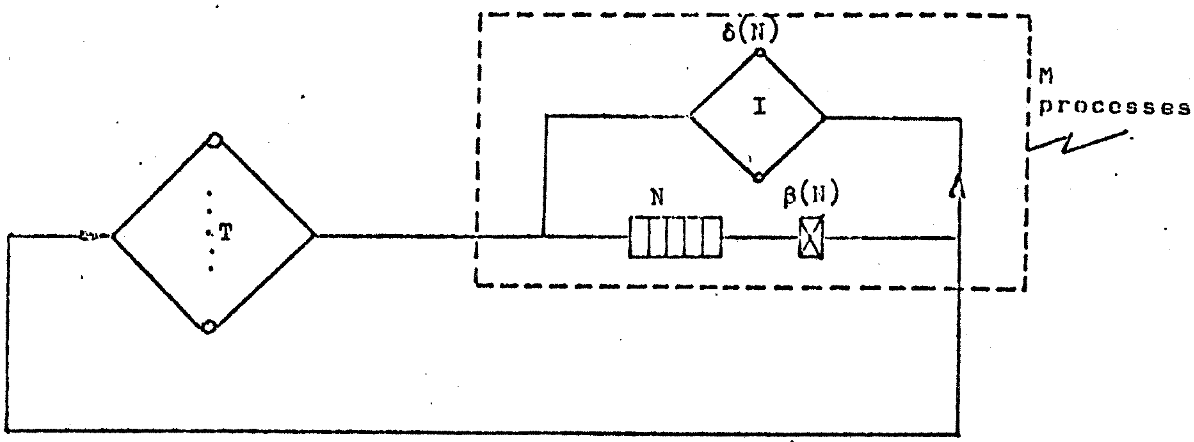


figure 3(a)

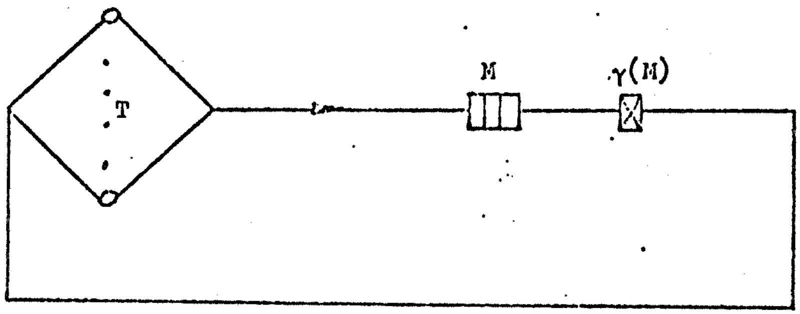


figure 3(b)

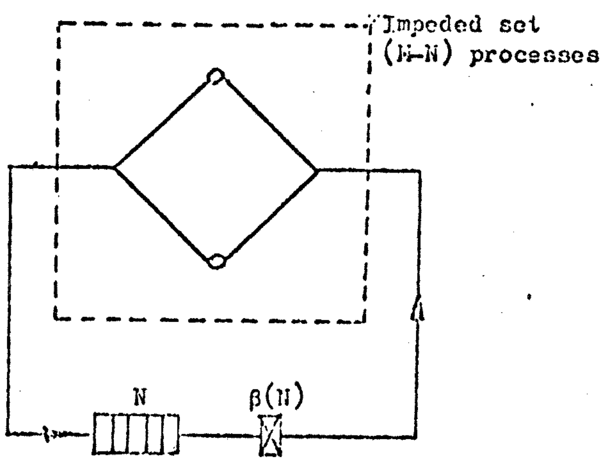


figure 3(c)

As a first step we will approximate the system of Figure 2(a) by the model shown on Figure 3(a). Let $\hat{\beta}(N,t)\Delta t$ be the probability that a process leaves the resource loop to enter either the <u>impeded</u> or <u>think</u> state in the interval $(t, t + \Delta t)$ when the number of processes in the resource loop is N. We may write

$$\hat{\beta}(N,t) = [x_{n+1}(N) + x_{n+2}(N)]\alpha(N)A(N,t)$$

where $A(N,t)$ is the probability that the CPU is not idle when there are N processes in the resource loop at time t. The approximation consists of replacing $A(N,t)$ by $A_0(N)$:

$$\hat{\beta}(N,t) \cong \beta(N) = [x_{n+1}(N) + x_{n+2}(N)]\alpha(N)A_0(N) \qquad (5)$$

where $A_0(N)$ is the stationary probability that the CPU queue in the closed model of Figure 2(b) (i.e. the resource loop with N processes and with no interactions with the terminals or with the impeded set) is not empty. This approximation will be valid [3,5] if for $1 \le i \le n$,

$$\begin{aligned} &\alpha(N)[x_{n+1}(N) + x_{n+2}(N)] << \mu_i \\ &h\beta(N) << \mu_i \ , \ h\beta(N) << \alpha(N)x_i(N) \qquad (6) \\ &\lambda(L-M) << \mu_i \ , \ \lambda(L-M) << \alpha(N)x_i(N) \end{aligned}$$

and if

$$x_{n+1}(N) + x_{n+2}(N) << x_i(N) \ , \ 1 \le i \le n. \qquad (7)$$

The second step is to approximate the model of Figure 3(a) by that of Figure 3(b) : the impeded set and the resource loop are replaced by a single server of service rate $\gamma(M)$ where M is the total number of processes in the system to the exception of those present at the terminals. By arguments similar to those yielding (5), we approximate $\hat{\gamma}(M,t)\Delta t$, the probability that a process returns to its terminal in $(t, t + \Delta t)$, by $\gamma(M)\Delta t$ where the approximation is valid if

$$\lambda(L-M) << \delta(N), x_{n+1}(N) >> x_{n+2}(N), \lambda(L-M) << \frac{x_{n+1}(N)}{x_{n+1}(N)+x_{n+2}(N)} \beta(N) \qquad (8)$$

$\gamma(M)$ is given by

$$\gamma(M) = \Sigma_{N=1}^{M} \frac{x_{n+2}}{x_{n+1} + x_{n+2}} \beta(N)p(N/M) \tag{9}$$

where $p(N/M)$ is the stationary probability of having N processes in the resource loop given that there are M processes in either the impeded set or in the resource loop supposing no interaction with the terminals for the subsystem of Figure 3(c).

After some computations, given in the Appendix, we obtain the expression

$$\gamma(M) = \left(\sum_{N=1}^{M} \frac{h^{N-1}}{A_0(N)(M-N)!} \right)^{-1} \sum_{N=1}^{M} \frac{h^{N-1}}{(M-N)!}$$

On Figure 4 we present some numerical values for $\gamma(M)$ as a function of M, h and of k (the exponent in the Belady life-time function in (3)). Here $n = 2$ (there are two peripheral units), and the parameter values used are given on the figure.

If k is small, i.e. when the effectiveness of the control is important, we see that for well-chosen values of h, $\gamma(M)$ does not vary appreciably as M increases which is the property we desire. However a more practical approach to understanding the effect of the random injection policy is to compare $\gamma(M)$ to the throughput of the resource loop $\beta^*(N)$ when the control is not applied where

$$\beta^*(N) \cong x_{n+2}(N)\alpha(N)A_0(N)$$

The comparison is given on Figure 5. However when k is relatively large (e.g. $k = 2.5$, on Figure 6) the system with no control has a better performance than the controlled system. The influence of the parameter h will be seen in the simulation experiments (Section 3) where it is evident that as long as h is sufficiently small, it can vary by an order of magnitude without affecting sensibly the behaviour of the controlled system.

Notice that in the theoretical model examined above the value of Q has no effect.

3. THE SIMULATION EXPERIMENTS

The theoretical analysis using the decomposition technique is a first-order approximation [5] when the inequalities (6) - (8)

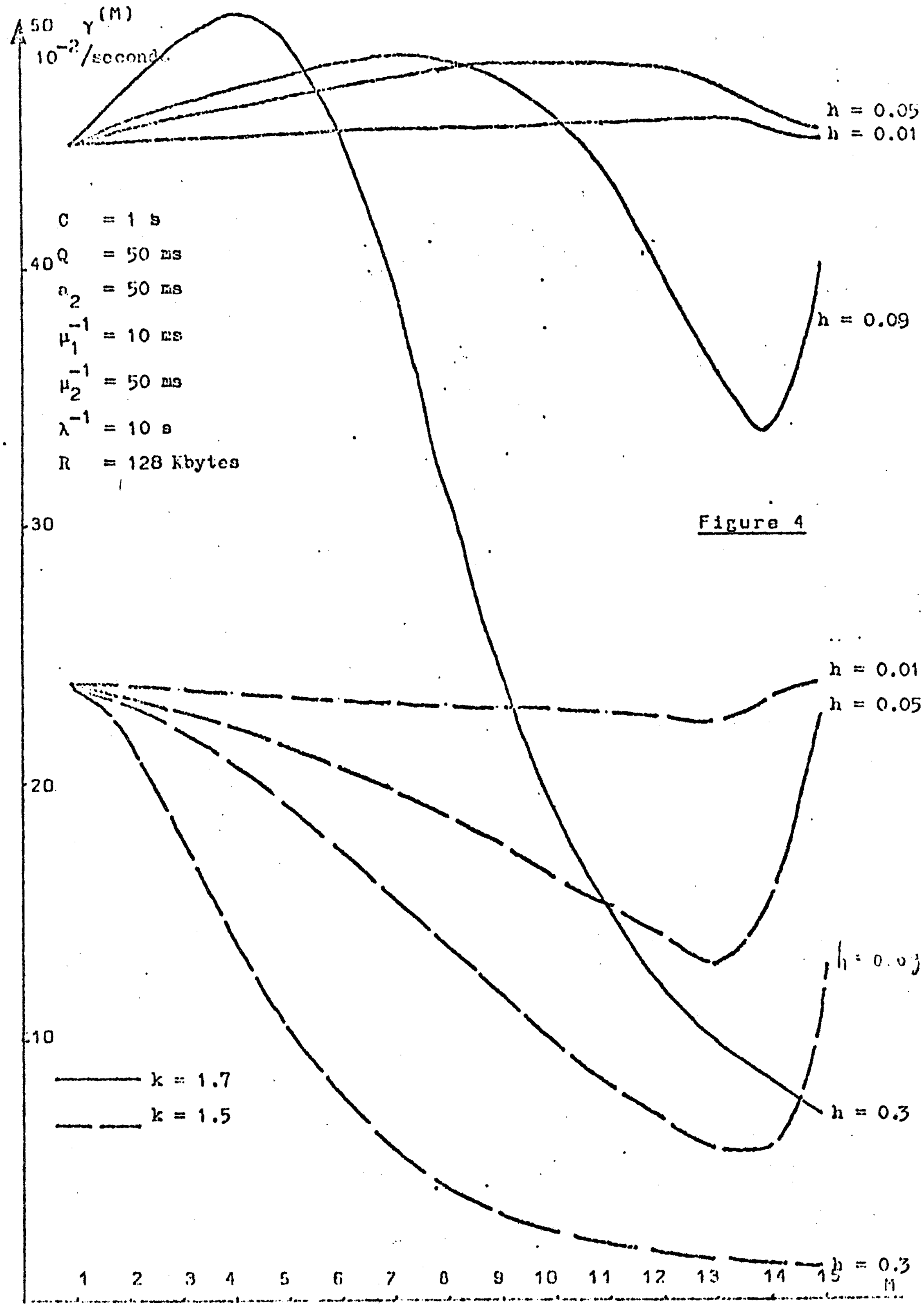
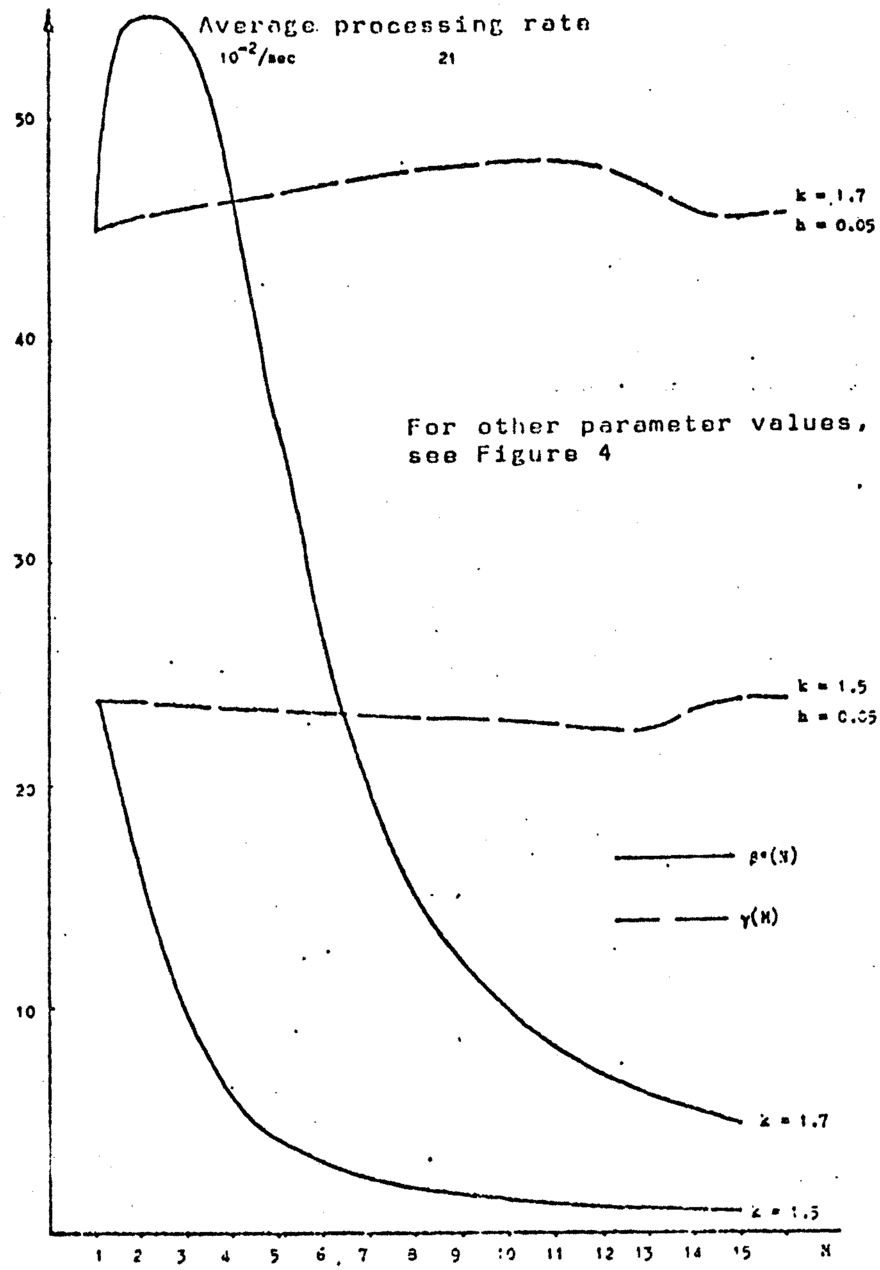

50 γ(M)
10^-2/second
C = 1 s
Q = 50 ms
a2 = 50 ms
μ1^-1 = 10 ms
μ2^-1 = 50 ms
λ^-1 = 10 s
R = 128 Kbytes
40
30
20
10
h = 0.05
h = 0.01
h = 0.09
h = 0.01
h = 0.05
h = 0.3
h = 0.3
k = 1.7
k = 1.5
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
M

Figure 4



Average processing rate
10^-2/sec
21
50
40
30
20
10
k = 1.7
h = 0.05
k = 1.5
h = 0.05
For other parameter values,
see Figure 4
β*(N)
γ(N)
k = 1.7
k = 1.5
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 N

Figure 5

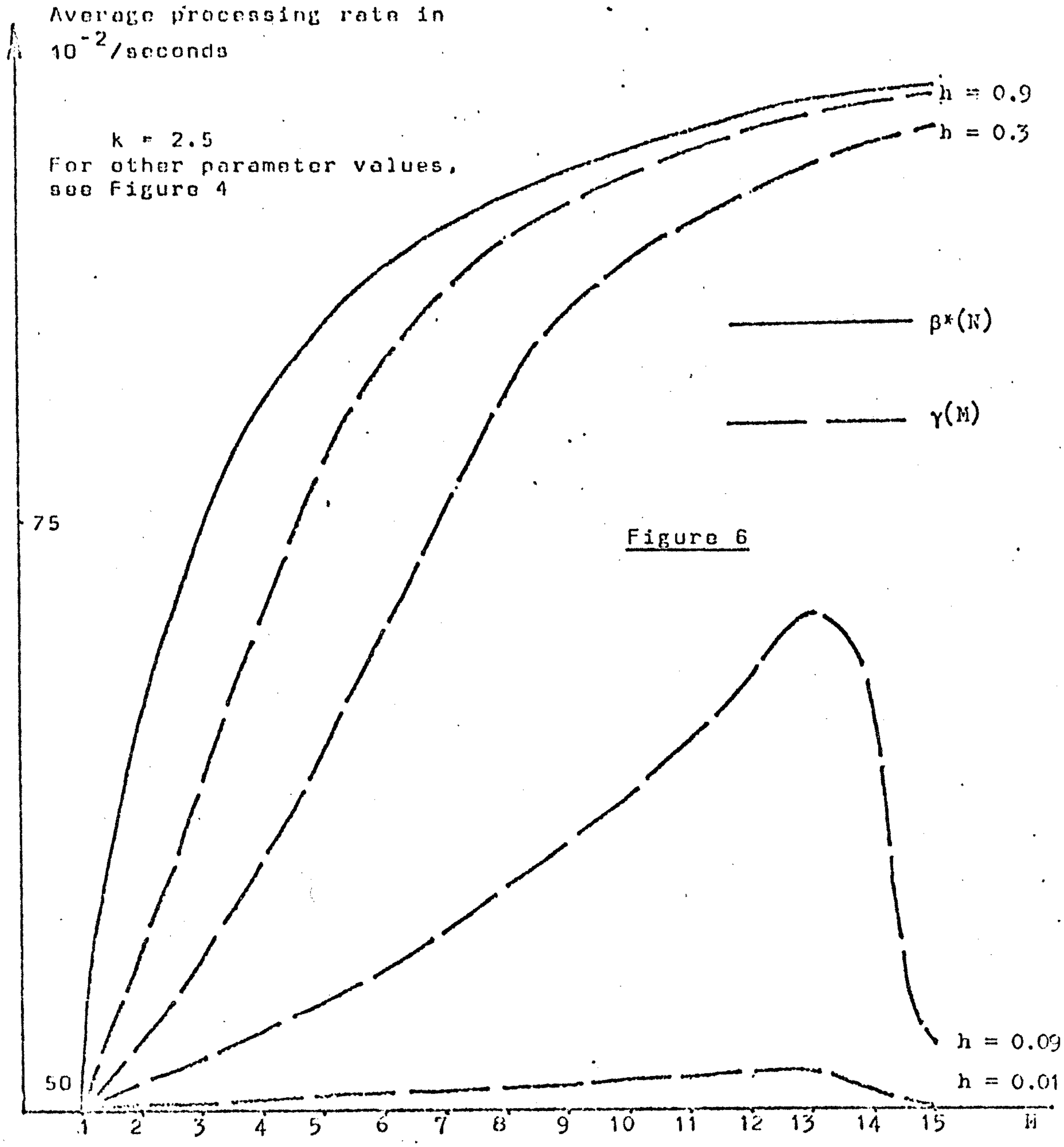


Figure 6

are satisfied. Thus it has to be verified whether the approximation is sufficiently good in the context of our model.

However the purpose of the simulation experiments we have conducted is not restricted to this point. We also use them to evaluate the following aspects :

- if the random injection control is not based on a continuous estimation of $\beta(N,t)$ does its effect still remain valid ?

- If the main distributions associated with the model are not exponential, is the mathematical model a robust predictor of system performance ?

The simulations will allow us to give affirmative replies to these two questions and also to investigate the effect of certain model parameters such as h or Q. We shall also compare the random injection control to the optimum control obtained by limiting the degree of multiprogramming.

3.1. Description of the simulator

The simulation model has been programmed with the following assumptions :

- the simulator corresponds to the system of Figure 3(a), where the value of $A_0(N)$ necessary to the computation of $\beta(N)$ is obtained by a separate simulator in which the service time at each peripheral device (P_1 and P_2) is constant (rather than exponential as in the mathematical model).
- in the simulation the total execution time of each process is constant (rather than exponential) of value C, and so is the execution time Q after which a process is placed into the impeded set.
- the random injection control in the simulator differs from that in the mathematical model. Since in practice it may be difficult to set $\hat{\delta}(N,t) = h\beta(N,t)$ at all times t, we have chosen to take a relatively unfavorable value for $\delta(N,t)$ in the simulator. If a process enters the impeded set at a time t' when the value of the degree of multiprogramming was $N_{t'}$, we allow it to remain in the impeded set for a random duration exponentially distributed of parameter $h\beta(N_{t'})$; thus the parameter used is not fully "up-to-date".

3.2. Confidence interval computations

For the computation of confidence intervals associated with the measurements made in the simulations we have used the classical method [12,13] of subdividing the measurements into continuous blocks such that the average values over each block of the parameter value are very weakly correlated. The block length chosen is of 25 measurements. However we have also computed confidence intervals without this scheme and the results are appreciably the same [1].

Although the primary performance measure which we are interested in is the average response time of a user, we do not give these measurements directly. In fact we observed that the confidence intervals associated with response time when directly measured were very large because of the relatively small number of samples. Therefore we measure the average number of processes which are not at the terminals as well as the related confidence interval, and the arrival rate of processes from the terminals. We then use these quantities in Little's formula $W.\lambda(L - E\{M\}) = E\{M\}$ in order to compute the estimated average response time $\hat{W} = \hat{M}/\lambda(L - \hat{M})$. The confidence interval of $\hat{W}$, (W_1, W_2) is obtained as follows : $W_1 = M_1/\lambda(L - M_1)$ and $W_2 = M_2/\lambda(L - M_2)$, where $\hat{M}$ is the estimated value of $E\{M\}$ and (M_1, M_2) is the corresponding confidence interval.

3.3. The simulation results

Comparison of the theoretical results developed in Section 2 with the simulations is given in Figures 7, 8, 9.

We see that except for large values of L the agreement is very good notwithstanding the important differences between the theoretical model and the simulation model. It is important to note that the 95 % confidence intervals are much larger for small k (Figure 7) than for larger values of k (Figures 8, 9) and the sensitivity of the confidence intervals to k is very large. On the other hand the sensitivity of the model to the distribution of the time spent in the impeded set is very small : on Figure 8 a constant time is compared with an exponentially distributed time spent in impeded state.

4. ADAPTIVE PROCEDURES FOR RANDOM INJECTION CONTROL (RIC)

In the previous sections no effort has been made to indi-

---

1) Through the purpose of our paper is not to comment on simulation methodology, we were not surprised by the identity of these results because of known properties of asymptotic normality of certain functions associated with Markov processes [14].

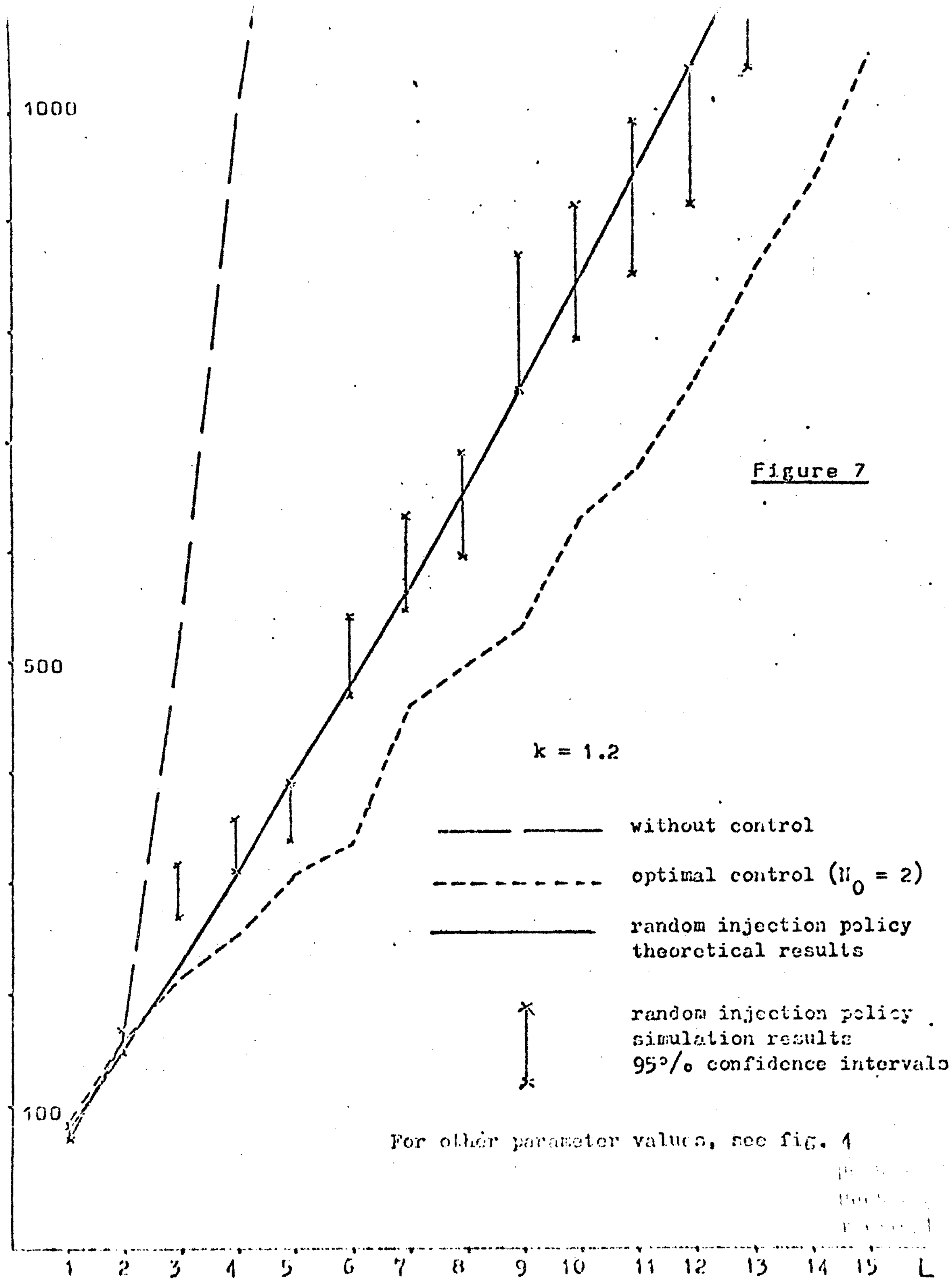

1000
500
100
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
L
k = 1.2
without control
optimal control (N_0 = 2)
random injection policy theoretical results
random injection policy simulation results 95°/o confidence intervals
For other parameter values, see fig. 4

Figure 7

cate specifically how the RIC method might be implemented in practice. This section is devoted to an evaluation of some possible approaches which we propose. Clearly, an approach similar in spirit to that developed in [7] may be applied. That is, an estimator of $A_0(N)$ may be used to determine N_0, the degree of multiprogramming which maximizes the throughput of the RL; one would then implement the optimal RIC discussed in the previous section. Another approach might be to use RIC with (1) giving the value of $\hat{\delta}(N,t)$, but taking

$$\hat{\beta}(N,t) = \hat{A}_0(N,t)(1/Q + 1/C)$$
$$\simeq \hat{A}_0(N,t)/Q$$

where $\hat{A}_0(N,t)$ is the estimated value of CPU utilisation at time t. These approaches will be discussed and evaluated in [15].

The approach we suggest and evaluate here makes use of an estimator of interdeparture times from the resource loop. Recall that RIC requires that $\delta(N,t) = h\beta(N,t)$. However in the implementation the rate $\delta(N,t)$ will be replaced by the corresponding inter-exit times from the impeded set; therefore we shall estimate these times directly.

Let $0 < t_1 < t_2 < \ldots < t_i < \ldots$ be the sequence of instants at which a customer leaves the RL to enter the impeded set or the terminals. Define the variable E_t as follows :

$$E_t = \begin{cases} 0 & \text{if} \quad t = 0 \\ E_{t_i} & \text{if} \quad t_i < t \leq t_{i+1} \\ E_{t_{i-1}}(1-\alpha) + \frac{\alpha}{h}(t_i - t_{i-1}) \end{cases} \qquad (11)$$

for some $0 \leq \alpha < 1$. For a process entering the impeded set at time t, we draw an exponentially distributed random variable X of expectation E_t so that the process will be reintroduced into the RL at time t + X. Notice that we need not have knowledge of N (the degree of multiprogramming) in order to implement this policy.Several simulation experiments have been conducted to evaluate this policy. The results are shown on Figures 12.

On Figure 12(a) we show the average response time of the system versus the total number of customers L with this control policy for k = 1.5 (the coefficient of locality in the Belady life-time functions). For a wide range of values of α(0.1 to 1) we see that the results remain very close to the theoretical results for RIC. The simple estimator (14) yields a control which improves considerably over system performance with no control. Similar results are obtained on Figure 12(b) with k = 1.8.

5. CONCLUSIONS

In this paper we have proposed a new method for the control of multiprogramming in a virtual memory system. The method has been justified using a mathematical model. A simulation model in which most probability distributions associated with the mathematical model have been modified then used to test the control scheme and the robustness of the model predictions.

Our results prove the interest of the random injection control policy for virtual memory systems. Of course, adaptive control procedures are valid only if the system overhead associated with the control is not unacceptably high. In a follow-up paper [15] we will examine the effect of overhead; we shall also introduce and analyse an optimal random injection control.

The theory predicts that Q, as we have chosen it, will have no effect on the average response time of the system; the simulations (see Figure 10) confirm that its effect is not significant on the <u>average</u> response time. Of course it will have an effect, similar to that of the quantum of a time-sharing discipline, on the response time of processes having longer or shorter execution time. In practice, one would use a smaller value of Q for those processes which have a high page fault rate and a larger Q for those have fewer page faults; this point will be discussed in [14].

On Figure 11 we have used simulation experiments to evaluate the effect of h. These results confirm the numerical predictions (see Figure 4). The small values of h are worse when L is small, since there we need little if any control of multiprogramming. The best value lies around h = 0.01 for large values of L, as indicated by the simulations (Figure 10) and the numerical studies (Figure 4).

APPENDIX

Outlined here are the details of the derivation of the resulting equivalent server (7). $\gamma(M)$ is given by

$$\gamma(M) = \Sigma_{N=1}^{M} \frac{x_{n+1}}{x_{n+1} + x_{n+2}} \beta(N) p(N/M) \tag{A1}$$

Recall that $p(N/M)$ is the steady state probability of having N processes in the resource loop given that there are M processes in either the impeded set or in the resource loop. We obtain this probability by solving the equilibrium equations for the system of Figure 3(c).

$$p(N/M)[\beta(N) + \delta(N)(M-N)] = p(N+1/M)\beta(N+1) + p(N-1/M)\delta(N)(M-N+1) \quad , \quad 1 \le N < M \tag{A2}$$

$$p(1/M)\beta(1) = p(0/M)\delta(0)M$$

$$p(M/M)\beta(M) = p(M-1/M)\delta(M-1)$$

The solution of these equations is

$$p(N/M) = p(0/M) \frac{\Pi_{i=0}^{N-1}\delta(i)(M-i)}{\Pi_{i=1}^{N}\beta(i)} \qquad 0 < N \le M \tag{A3}$$

and we obtain $p(0/M)$ by using the condition

$$\Sigma_{N=0}^{M} p(N/M) = 1$$

This yields the result :

$$p(0/M) = [\Sigma_{N=1}^{M} \frac{\Pi_{i=0}^{N-1}\delta(i)(M-i)}{\Pi_{i=1}^{N}\beta(i)} + 1]^{-1} \tag{A4}$$

Using this value, and (1), expressions (A2) and (A3) become

$$p(N/M) = p(0/M) \frac{h^{N-1}B}{\beta(N)} \Pi_{i=0}^{N-1} (M-i) \qquad 1 \le N \le M \tag{A5}$$

$$p(0/M) = [\Sigma_{N=1}^{M} \frac{h^{N-1}B}{\beta(N)} \Pi_{i=0}^{N-1} (M-i) + 1]^{-1} \quad (A5)$$

and replace p(N/M) by its value in (A1) :

$$\gamma(M) = Bp(0/M)M! \, \Sigma_{N=1}^{M} \frac{x_{n+1}}{x_{n+2} + x_{n+1}} \frac{h^{N-1}}{(M-N)!}$$

$$p(0/M) = (1 + BM! \, \Sigma_{N=1}^{M} \frac{h^{N-1}}{\beta(N)(M-N)!})^{-1} \quad (A6)$$

We know that

$$\frac{x_{n+1}}{x_{n+2} + x_{n+1}} = \frac{1/C}{1/C + 1/Q}$$

Let B tend to infinity. The limit of (A6) is

$$\lim_{B\to\infty} \gamma(N) = 1/C \, \Sigma_{N=1}^{M} \frac{h^{N-1}}{(M-N)!} \, (\Sigma_{N=1}^{M} \frac{h^{N-1}}{A_0(N)(M-N)!})^{-1}$$

which is the expression given in (7).

Notice that we set B→∞ (as mentioned in the discussion following equation (1)) to represent the fact that when N=0, no time is spent by processes in the impeded set.

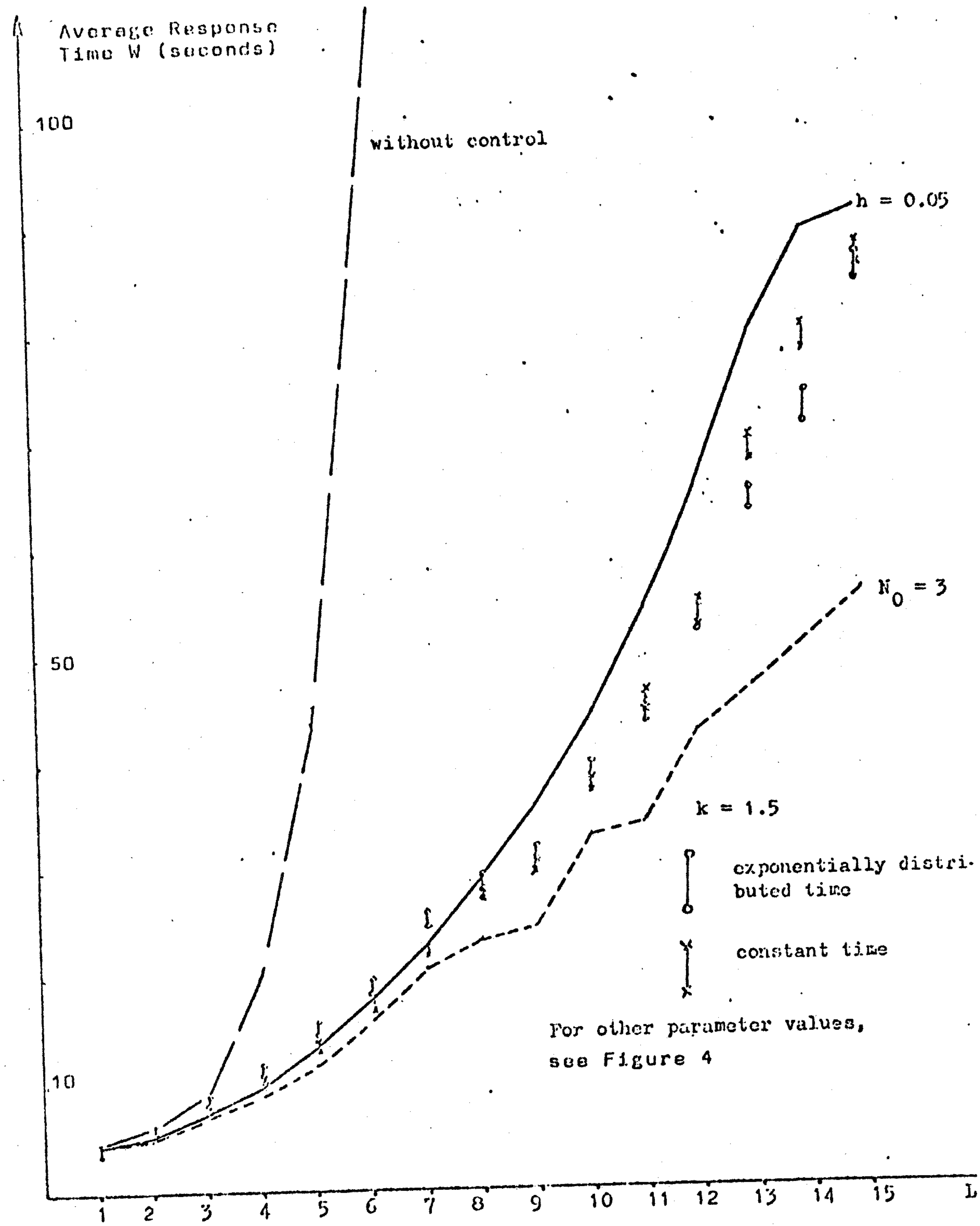

Average Response
Time W (seconds)
100
50
10
without control
h = 0.05
N_0 = 3
k = 1.5
exponentially distri-
buted time
constant time
For other parameter values,
see Figure 4
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
L

figure 8

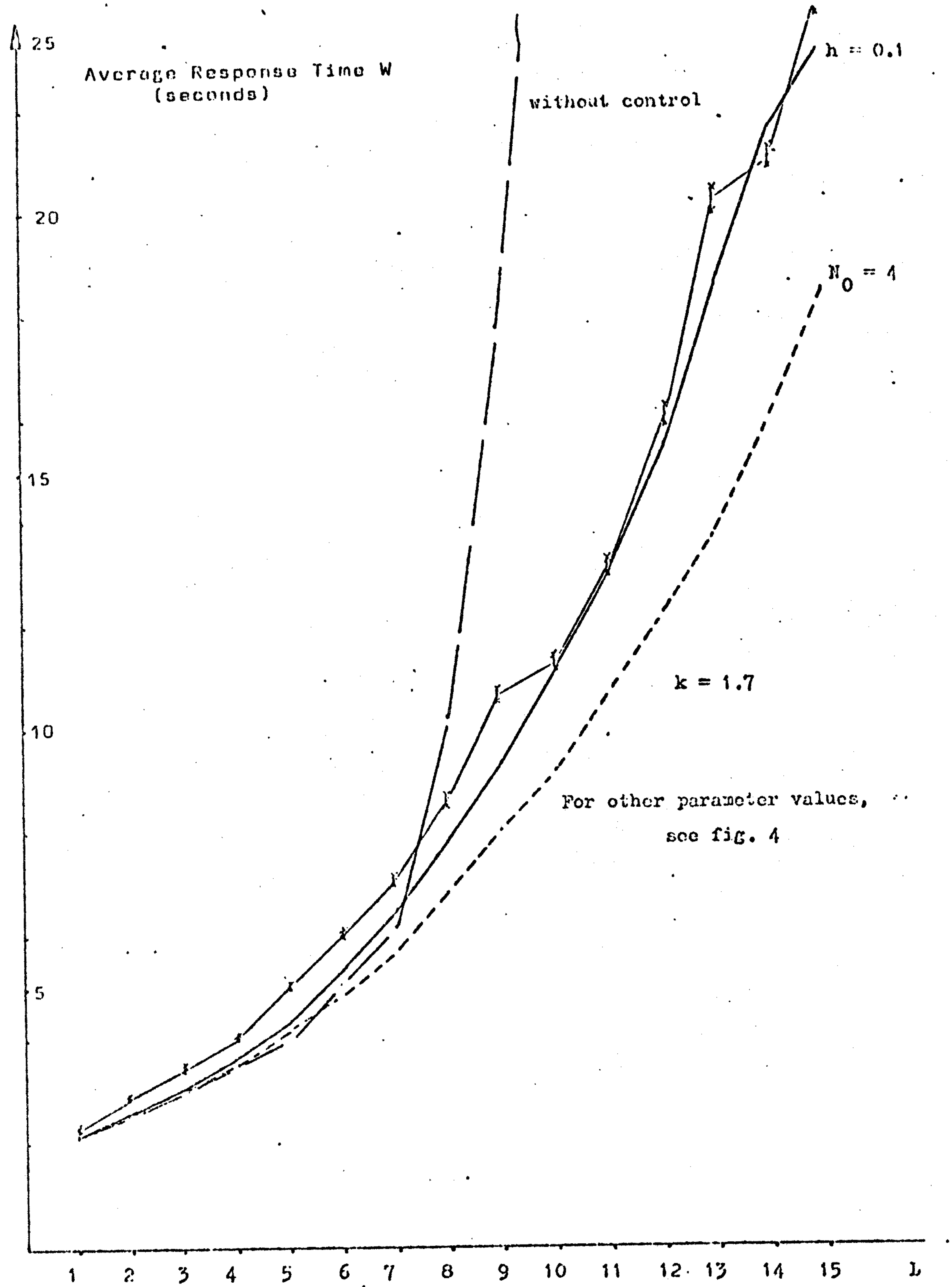

25
Average Response Time W
(seconds)
without control
h = 0.1
20
N_0 = 4
15
k = 1.7
10
For other parameter values,
see fig. 4
5
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
L

figure 9

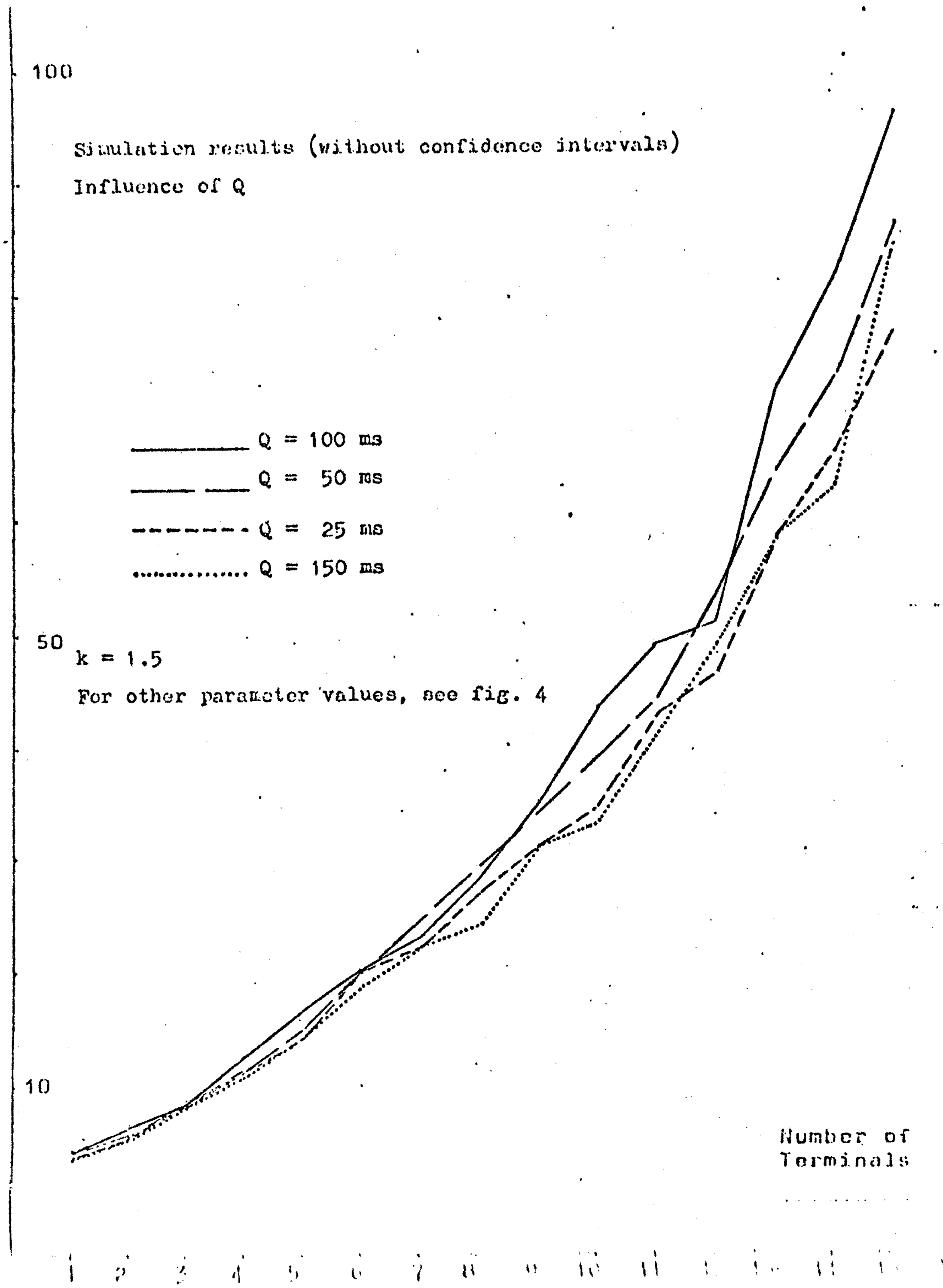
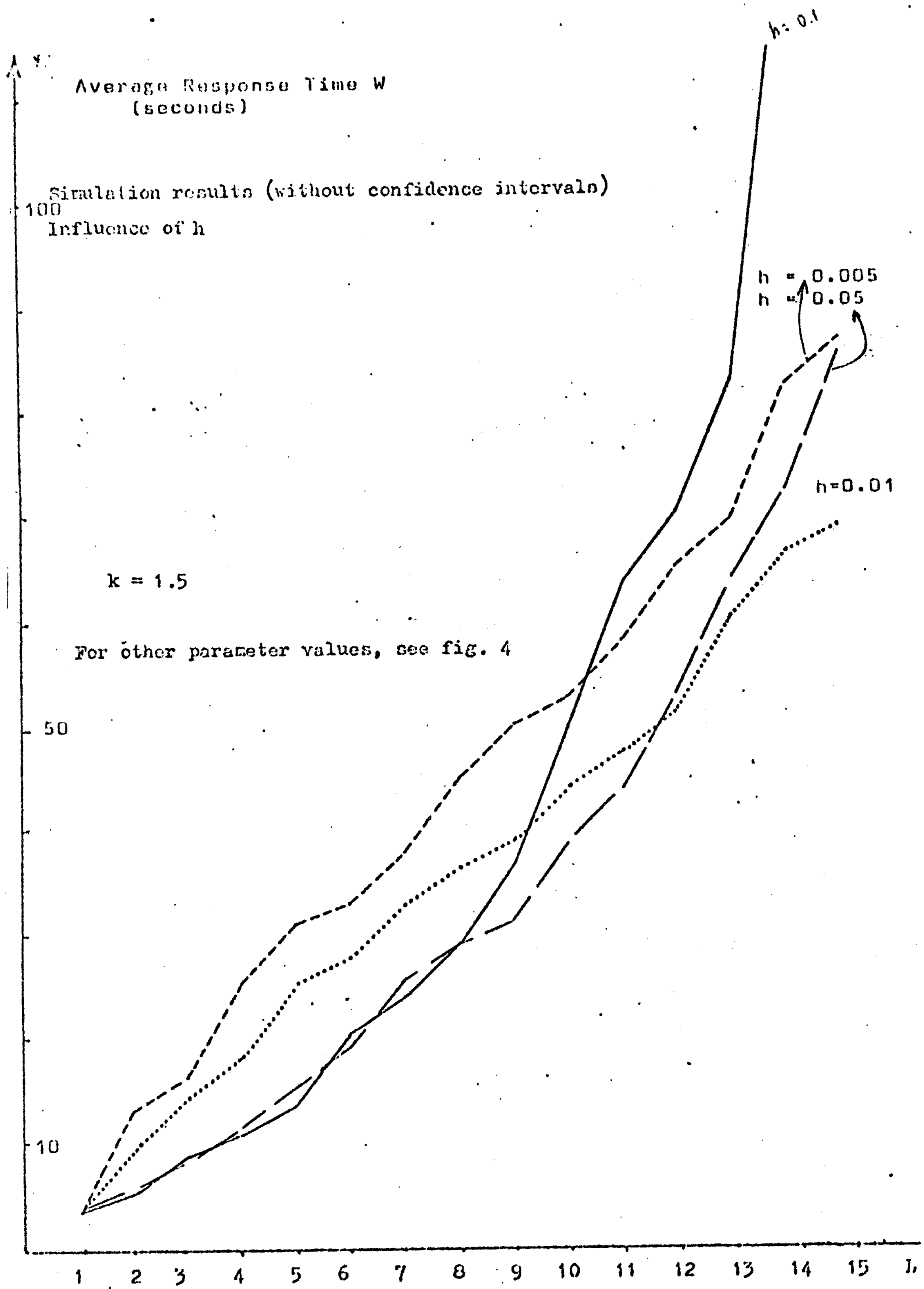

100
Simulation results (without confidence intervals)
Influence of Q
Q = 100 ms
Q = 50 ms
Q = 25 ms
Q = 150 ms
50
k = 1.5
For other parameter values, see fig. 4
10
Number of Terminals

Figure 10



h: 0.1
Average Response Time W
(seconds)
Simulation results (without confidence intervals)
100
Influence of h
h = 0.005
h = 0.05
h=0.01
k = 1.5
For other parameter values, see fig. 4
50
10
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
L

Figure 11

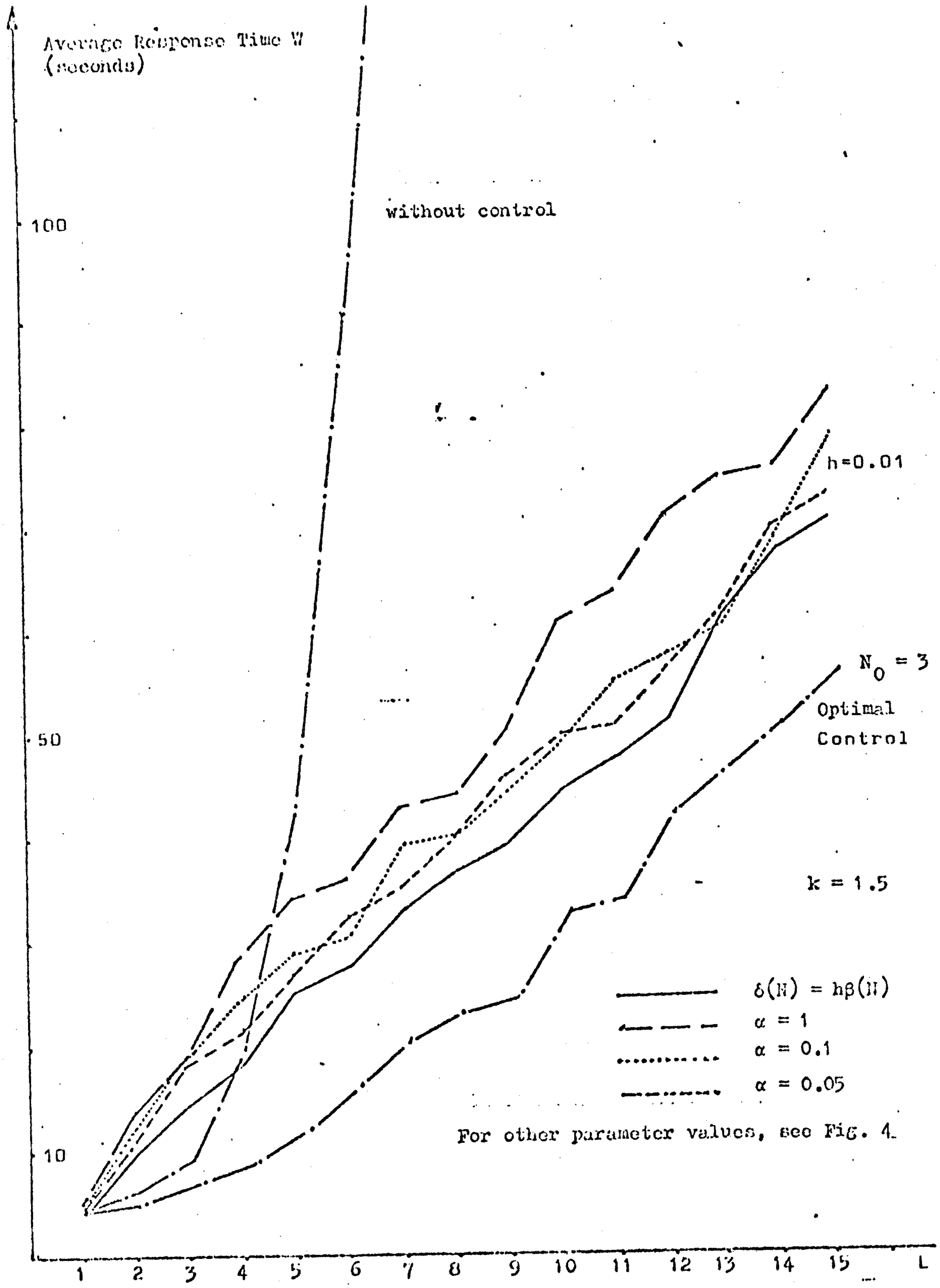

Average Response Time W
(seconds)
100
50
10
without control
h=0.01
N_0 = 3
Optimal
Control
k = 1.5
δ(N) = hβ(N)
α = 1
α = 0.1
α = 0.05
For other parameter values, see Fig. 4.
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
L

Figure 12(a)

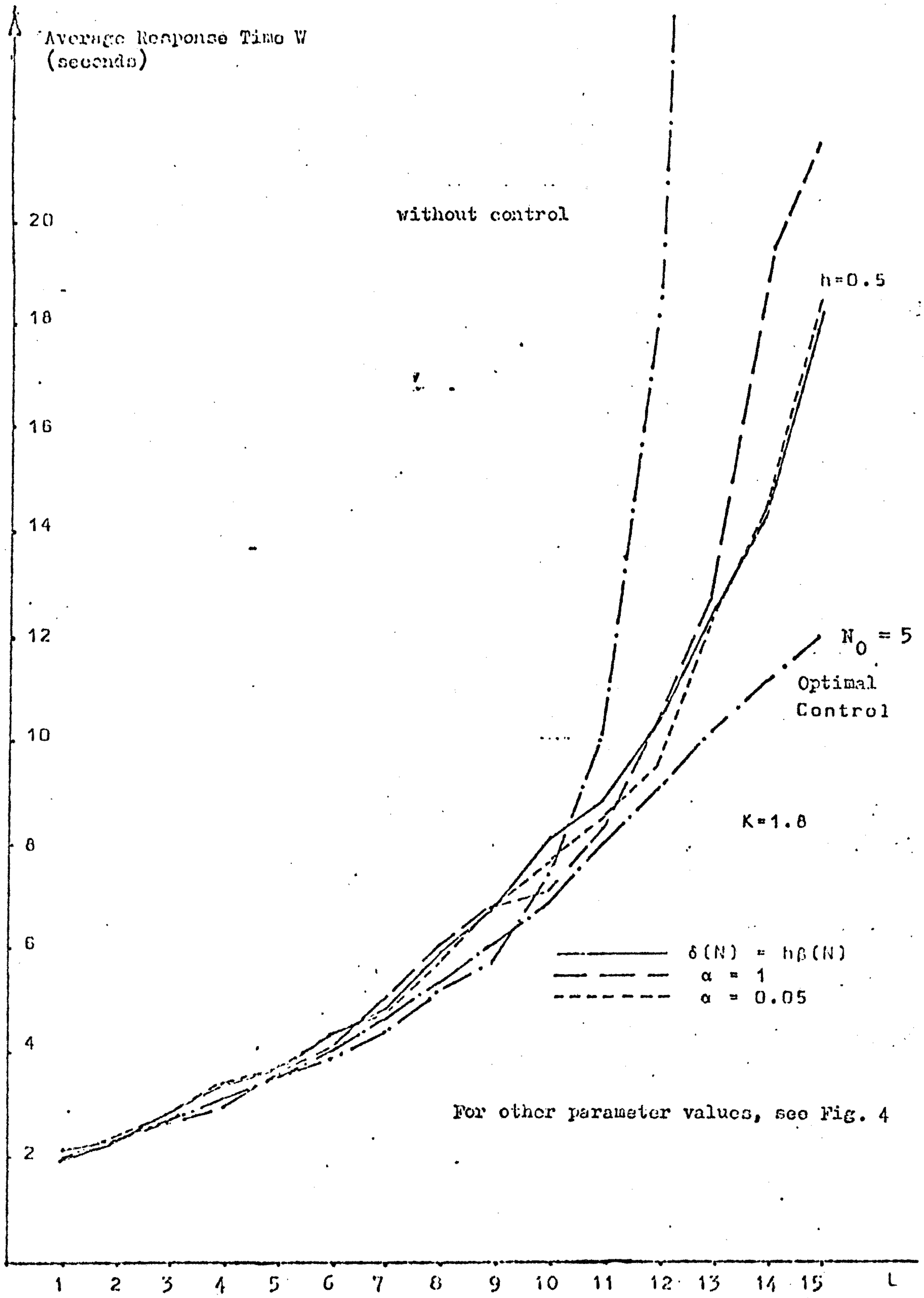

Average Response Time W
(seconds)
20
18
16
14
12
10
8
6
4
2
without control
h=0.5
N_0 = 5
Optimal
Control
K=1.8
δ(N) = hβ(N)
α = 1
α = 0.05
For other parameter values, see Fig. 4
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
L

Figure 12(b)

REFERENCES

[1] L. BELADY, C.J. KUEHNER, "Dynamic Space Sharing in Computer Systems", CACM 12, 5, May (1969), 282-288.

[2] P. DENNING, 'Thrashing : Its Causes and Prevention", Proc. AFIPS 68, Fall Joint Comput. Conf., 33, 915)922.

[3] P.J. COURTOIS, "On the Near Complete Decomposability of Networks of Queues and of Stochastic Models of Multiprogramming Systems", Carnegie-Mellon University, Computer Science Dept., Research Report, November (1971).

[4] P.J. COURTOIS, "Decomposability, Instabilities, and Saturation in Multiprogramming Systems", CACM 18, 7, July (1975), 371-377.

[5] P.J. COURTOIS, "Error Analysis in Nearly-completely Decomposable Stochastic Systems", Econometrica, 43, 4, July (1975), 691-709.

[6] A. BRANDWAJN, J. BUZEN, E. GELENBE, D. POTIER, "A Model of Performance for Virtual Memory Systems", Proc. ACM-SIGMETRICS Symposium, Montreal, September (1974), to appear in IEEE Trans. Software Engineering.

[7] M. BADEL, E. GELENBE, J. LEROUDIER, D. POTIER, "Adaptive Optimization of a Virtual Memory System", Proc. IEEE, Special Issue on Interactive Systems, June (1975).

[8] P. DENNING, S. GRAHAM, "Multiprogrammed Memory Management", Proc. IEEE, Special Issue on Interactive Systems, June (1975).

[9] P. DENNING, S. GRAHAM, "Multiprogramming and Program Behaviour" Proc. ACM-SIGMETRICS, 3, 4, September (1974).

[10] J. LEROUDIER, D. POTIER, "Principles of Optimality for Multiprogramming", Intern. Symp. on Computer Performance, Modelling, Measurements, Evaluation, Harvard, March 19, (1976).

[11] E. GELENBE, R.R. MUNTZ, "Exact and Approximate Solutions to Probabilistic Models of Computer System Behaviour - Part I", to appear in Acta Informatica.

[12] J. LEROUDIER, M. PARENT, "Quelques aspects de la modélisation des systèmes informatiques par simulation à événements discrets", RAIRO, Informatique, Vol. 10, n° 1, Janvier (1976), p. 5 à 26.

[13] R.W. CONWAY, "Some Tactical Problems in Digital Simulation", Management Sciences, 10, 1, October (1963).

[14] P. LE GALL, "Convergence des Simulations et Application aux Réseaux Téléphoniques", AFIRO monographies de recherche opérationnelle, 7, (1968).

[15] E. GELENBE, A. KURINCKX, "Optimal and Sub-optimal Random Injection Control of Multiprogramming", in preparation.

STABILITY AND CONTROL OF PACKET-SWITCHING BROADCAST CHANNELS

G. Fayolle*, E. Gelenbe**, J. Labetoulle*

ABSTRACT

In this paper we consider the behaviour of the slotted broadcast channel used by an ensemble of terminals for the transmission of packets of data. A mathematical model is used to prove that the channel is unstable, leading to zero effective throughput, if no control is imposed on the channel behaviour. Two classes of control policies, acting on the input and on retransmissions from blocked terminals, are then analyzed and stability and optimality conditions for the channel with these policies are derived. The theorem on instability of the uncontrolled channel can in fact be considered as being a corollary of the stability theorem for the retransmission control policy. We show that e^{-1} is a lower bound to the maximum achievable throughput with an input control policy. On the other hand, an optimal retransmission control policy must regulate retransmissions so that the probability of retransmission of an individual blocked terminal is of the form $(1-\lambda)n^{-1}$ in each slot, where n is the total number of blocked terminals. Some simulation results are provided in order to illustrate the effect of this policy.

* *IRIA-LABORIA*

** *Chaire d'Informatique de l'Université de Liège et IRIA-LABORIA.*

1. INTRODUCTION.

Computer networks using packet-switching techniques have been implemented [1,2,3,4,5,10] in order to allow a large community of communicating users to share and transmit data and to utilize excess computing power which may be available at remote locations in an efficient manner. In this paper, we shall be concerned with packet switching networks using radio channels similar to the ALOHA system [1].

We consider a large set of terminals communicating over a single radio channel in such a way that a packet is successfully transmitted only if its transmission does not overlap in time with the transmission of another packet; otherwise all packets being simultaneously transmitted are lost. A terminal whose transmission is unsuccessful is said to be blocked; it has to repeat the transmission until it achieves success. A terminal which is not blocked is either active or it is transmitting a packet. The operation of the system is shown schematically in Figure 1 where the different state transitions of a terminal are shown. Since the only means of communication between terminals is the channel itself, it is not easy to schedule transmissions so as to avoid collisions between packets. It is also abvious that a terminal would in no case transmit more than one packet simultaneously.

Various methods for controlling the transmission of packets have been suggested. The simplest is to allow terminals to transmit packets at any instant of time. The second method, known as the slotted ALOHA scheme has been shown to increase channel throughput over the first method [6]. Here time is divided into "slots" of equal duration; each slot can accomodate the transmission time of one packet and packets are all of the same length. Packet transmission is synchronized so as to be initiated at the beginning of a slot for any terminal and it terminates at the end of the same slot. Other schemes have been suggested elsewhere [9].

KLEINROCK and LAM [8] have discussed the stability problem of the slotted ALOHA channel. They give qualitative arguments and results based on simulations indicating that the channel becomes saturated if the set of terminals is very large, independently of the arrival rate of packets to the channel, saturation being the phenomenon whereby the number of blocked terminals becomes very (or arbitrarily) large. They also compute the expected time to attain a given level of saturation. In [11] policies designed to optimize the throughput of the channel, defined as the expected number of successful transmissions per slot, are presented.

The purpose of this paper is to give a theoretical treatment of some control policies which can be applied to the broadcast channel in order to stabilize it and to maximize its performance. We first recall the proof of instability in [12] extending it to the finite source model taken in the limit as the total number of terminals becomes very large, and showing that channel instability implies that the equilibrium value of the throughput is zero. Two simple control policies are then presented and necessary and sufficient conditions for stability of the controlled channel are derived. Bounds for the equilibrium value of the channel throughput with these policies are obtained. Finally we give a simple algorithm for the approximate implementation of this policy and exhibit some simulation results showing its performance.

2. A MATHEMATICAL MODEL.

A precise definition of stability can only be considered in the context of a model of the behaviour of the broadcast channel. In this section we present a model identical ot the one we have considered in an earlier paper [12] except that we shall take into account here both finite and infinite source systems.

Assuming that the slot, and the time necessary to transmit a packet, are of unit length, consider N(k) the number of blocked terminals at the instants k = 0,1,2,... when a slot begins. Let X_k be the number of packets transmitted by the set of active terminals during the k-th slot and denote by Y_k the number of blocked terminals transmitting during the k-th slot. In the *infinite source model* (X_k) is the sequence of independent and identically distributed random variables with common distribution given by

(1) $$\Pr(X_k = i) = c_i \quad , \; i \geq 0$$

In the *finite source model* assuming that the total number of terminals in the system is M, we let the event $(X_k = i/N(k)=j)$ be independent of values of X_t for $t < k$; its probability is given by

(2) $$q_j(n) = \Pr(X_k = j/N(k)=n) = \binom{M-n}{j} b^j (1-b)^{M-n-j}$$

for $0 \leq j \leq M-n$, where b is the probability that any one active terminal transmits a packet during a slot.

For both models we shall denote by f the probability that any one blocked terminal transmits a packet during a slot. We then define

(3) $$g_i(n) = \Pr(Y_k = i/N(k)=n)$$

where we assume that the event $(Y_k/N(k))$ is independent of Y_t for $t < k$. Therefore

$$(4) \qquad g_i(n) = \binom{n}{i} f^i (1-f)^{n-i}$$

and more particularly

$$(5) \qquad g_0(n) = (1-f)^n \ , \ g_1(n) = nf(1-f)^{n-1}$$

Definition 1.

The infinite source broadcast channel is *unstable* if for $k \to \infty$ the probability $Pr(N(k) < j) \to 0$ for all finite values of j; otherwise it is *stable*. For the finite source model, the system is unstable if the above condition is verified as we let $M \to \infty$, $b \to 0$, $M.b \to d$, where d is a constant.

The definition given here simply states that instability is verified if (with probability one) the number of blocked terminals becomes infinite as time tends to infinity.

Theorem 1.

The broadcast channel is unstable both for the finite and infinite source model.

Proof.

Let us first consider the infinite source model. The proof given here is identical to the one we presented in [12]. Let $p_n(k)$ denote the probability that $N(k) = n$. The following transition equation may be written for the infinite source model (1):

---

(1) Equation (6) is valid for all $n \geq 0$ if we adopt the rule that $p_i(k) = 0, i < 0$.

$$(6) \qquad p_n(k+1) = \sum_{j=2}^{n} p_{n-j}(k)c_j + p_{n+1}(k)g_1(n+1)c_0 + p_n(k)(1-g_1(n))c_0 + p_n(k)g_0(n)c_1 + p_{n-1}(k)(1-g_0(n-1))c_1$$

On the right-hand side of (6), the first term covers the cases where two or more packets have been transmitted by the active terminals during the k-th slot; the second term covers the case in which exactly one blocked terminal has transmitted while no active terminal has done so. Notice that $\{N(k);k=0,1,\ldots\}$ is a Markov chain and that it is aperiodic and irreducible. It is ergodic if an invariant probability measure $\{p_n;n=0,1,\ldots\}$ exists satisfying (6) such that $p_n > 0$ for all n and where $p_n = \lim_{k\to\infty} p_n(k)$. To show that $\lim_{k\to\infty} \Pr(N(k) < j) = 0$ for all finite values of j it suffices that the Markov chain representing the number of blocked terminals be not ergodic. Substituting p_n for $p_n(k)$ and $p_n(k+1)$ in (6) we obtain

$$(7) \quad p_n = \sum_{j=0}^{n} p_{n-j}c_j + p_{n+1}g_1(n+1)c_0 + p_n(g_0(n)c_1 - g_1(n)c_0) - p_{n-1}g_0(n-1)c_1$$

Let

$$(8) \qquad S_N = \sum_{n=0}^{N} p_n$$

we then have for any $N \geq 0$,

$$(9) \qquad S_N = p_{N+1}g_1(N+1)c_0 + p_N g_0(N)c_1 + \sum_{n=0}^{N} S_{N-n}c_n$$

or

$$(10) \qquad S_N(1-c_0) = \sum_{n=1}^{N} S_{N-n}c_n + p_{N+1}g_1(N+1)c_0 + p_N g_0(N)c_1$$

or equivalently

$$(11) \qquad p_N(1-c_0) \le p_{N+1}g_1(N+1)c_0 + p_N g_0(N)c_1$$

But then, from (5) and (11) we have

$$(12) \qquad \frac{p_{N+1}}{p_N} \geqslant \frac{1 - c_0 - (1-f)^N c_1}{(N+1)f(1-f)^N c_0}$$

for any non-negative integer N. This implies that the ratio $(p_{N+1}/p_N) \to \infty$ as $N \to \infty$, so that the sum S_∞ can only exist if $p_N=0$ for all finite values of N ; otherwise S_∞ is divergent and this cannot be the case since the p_N, $N \ge 0$, define a probability distribution. Thus the Markov chain representing the number of blocked terminals is not ergodic, and the broadcast channel under the infinite source assumption is unstable.

Now consider the finite source model. Using the rule that $p_i(k)=0$ for $i < 0$, the transition equation for $0 \le n < M$ is

$$(13) \qquad p_n(k+1) = \sum_{j=2}^{n} p_{n-j}(k)q_j(n-j) + p_{n+1}(k)g_1(n+1)q_0(n+1) + p_n(k)(1-g_1(n))q_0(n) + p_n(k)g_0(n)q_1(n) + p_{n-1}(k)(1-g_0(n-1))q_1(n-1)$$

Defining, for $0 \le N < M$, the sum S_N as in (8) for the finite source model we obtain from (13) and substituting the stationary probability p_n :

$$(14) \quad S_N = p_{N+1}g_1(N+1)q_0(N+1)+p_N g_0(N)q_1(N) + \sum_{n=0}^{N} \sum_{j=0}^{n} p_{n-j}q_j(n-j)$$

Now take limit as in Definition 1 : $M\to\infty$, $b\to 0$, $M.b\to d$; we obtain $q_j(n) = \frac{d^j}{j!} e^{-d}$ for any j and n. Therefore, in the limit

$$(15) \quad S_N = p_{N+1}g_1(N+1)q_0(N+1) + p_N g_0(N)q_1(N) + \sum_{j=0}^{N} \frac{d^j e^{-d}}{j!} S_{N-j}$$

and an argument identical to the one for the infinite source model can be now used to complete the proof of instability.

We note in passing that the finite source model in the limit as we let the total number of terminals tend to infinity, and the infinite source model are not identical; in the infinite model there is a non-zero probability of a transmission from active terminals in each slot even when we let $k\to\infty$, while for the finite source model in the limit as $M\to\infty$ no active terminal will transmit as $k\to\infty$.

In the context of this study, another measure of interest is the throughput of the broadcast channel. Indeed this may well be the primary performance measure for the system under consideration.

Definition 2.

The *conditional throughput* $D_n(k)$ of the broadcast channel is the conditional probability that one packet is successfully transmitted during the k-th slot given that $N(k)=n$.

Clearly, the conditional throughput cannot exceed one; it can also be defined as the expected value of the number of successful transmissions during the k-th slot conditional on there being n blocked terminals at the beginning of that slot.

Definition 3.

The *throughput* of the brodcast channel is defined as

$$D = \lim_{k\to\infty} \sum_{n=0}^{\infty} D_n(k)p_n(k)$$

The conditional throughput is

(17) $$D_n(k) = c_0 g_1(n) + c_1 g_0(n)$$

for the infinite source model; for the finite source model we replace c_0 and c_1 by $q_0(n)$ and $q_1(n)$, respectively. This quantity is obviously independent of k, therefore in the following we shall simply write D_n instead of $D_n(k)$.

Theorem 2.

For $f > 0$, the throughput of the broadcast channel is zero for the infinite source model, and for the finite source model as we let $M\to\infty$, $b\to 0$, $M.b\to d$.

The proof is straighforward and not presented here.

3. CERTAIN CHANNEL CONTROL POLICIES.

Various control policies for the broadcast channel have been discussed in [11] where these have been classified, roughly speaking, into three groups : policies which regulate access to the channel from the active terminals, those which regulate access from the blocked terminals, and mixed policies. In this section we discuss two policies in some detail and give a definition of stability in each case. We see that this definition will be a variant of (or identical to) the definition given above. The first control policy which we shall describe typifies the first group of policies and it may well be impossible to implement; the second policy is of the second group and has a better chance of being realizable.

3.1. *A threshold control policy*.

An input control policy as defined by LAM [11] is one which limits access to the channel from the active terminals depending on the present state and past history of the channel. Borrowing the terminology of Markov decision theory [13] , a policy is said to be *stationary* if it only depends on the present state of the system.

The first policy we present is described in Figure 2. If the number of blocked terminals exceeds Θ , the threshold, an active terminal which wishes to initiate the transmission of a packet is not allowed to transmit and joins the *impeded set*; if not the transmission takes place as in the uncontrolled channel. As soon as the number of blocked terminals decreases below Θ (this can only take place in steps of one) an impeded terminal joins the blocked set; thus the number of blocked terminals can be less than Θ only if there are no impeded terminals. The retransmission rate of blocked terminals is constant. We shall refer of this scheme as the *threshold control policy*.

In this context, stability must be defined in terms of the number of impeded plus blocked terminals.

Definition 4.

Let $U(k)$ be the number of blocked plus impeded terminals at the beginning of the k-th slot for the threshold control policy. The channel, with this control scheme, is unstable if the limit as $k \to \infty$ of $Pr\{U(k) < j\}$ is zero for all finite values of j for the infinite source model; for the finite source model the same definition is used as $M\to\infty$, $b\to 0$, $M.b\to d$. Otherwise the channel is stable.

The following equations, which must be satisfied by the equilibrium probabilities p_n for the number n of blocked plus impeded terminals at the beginning of a slot, may be derived.

$n \leq \Theta$

$$(19)\quad p_n = \sum_{j=0}^{n} p_{n-j}c_j + p_{n+1}A_1(n+1)c_0 + p_n[c_1 g_0(n) - c_0 g_1(n)] - p_{n-1}g_0(n-1)c_1$$

$n \geq \Theta + 1$

$$(20)\quad p_n = \sum_{i=n-\Theta}^{n} p_{n-i}c_i + \sum_{i=1}^{n-\Theta-1} p_{n-i}c_i[1-A_1(n-i)]$$

$$+ \sum_{i=1}^{n-\Theta} p_{n-i+1}c_iA_1(n-i+1) + p_{n+1}c_0A_1(n+1) + p_n[1-A_1(n)]c_0$$

where

$$A_1(n) = \begin{cases} g_1(\Theta) & \text{if } n > \Theta \\ g_1(n) & \text{if } 0 \leq n \leq \Theta \end{cases}$$

Equation (20) may be rewritten as :

$$(21)\quad p_n = \sum_{i=0}^{n} p_{n-i}c_i + \sum_{i=0}^{n-\Theta-1} p_{n-i}A_1(n-i)[c_{i+1}-c_i] + p_{n+1}c_0A_1(n+1)$$

We obtain the following result concerning the stability of the treshold control policy. For simplicity let $A = g_1(\Theta)$.

Theorem 3.

If the expected arrival rate of active packets $\lambda = \sum_{i=1}^{\infty} i c_i$ for the infinite source model is less than A, then the broadcast channel with a stationary threshold control policy is stable; otherwise it is unstable.

The proof is given in Appendix 1.

The threshold control policy may be quite difficult to implement in practice. It has a major advantage, however, with respect to the retransmission control policies we shall study in Section 3.2 : the maximum achievable channel throughput is *not* limited to e^{-1}. In fact the throughput may be arbitrarily close to one if $\Theta = 1$ since it suffices to set $f = 1$ in this case. In general, for $\Theta \geq 1$, A is maximized by setting f equal to $f^* = \Theta^{-1}$. We then have $A(f^*) = (1-\Theta^{-1})^{\Theta-1}$ which, for $\Theta >> 1$ is $A(f^*) \simeq \exp(-1+\Theta^{-1}) > e^{-1}$. We see here that e^{-1} is a *lower bound* to the maximum achievable throughput. This does not depend on the Poisson assumption of packet arrivals to the channel.

3.2. *A retransmission control policy*.

A retransmission control policy is one which regulates access to the channel from the set of blocked terminals as a function of the past and present state of the system. We consider such a policy which only uses information concerning the present state (it is stationary) to regulate the retransmission rate of the ensemble of blocked terminals. The appropriate definition of stability (for this case) is then that given in Definition 1 and the equations for the controlled system are (6) for the infinite source model and (7) for the finite source model with the following modification. The parameter f which determines $g_i(n)$ (see (3) and (4)) giving the probability that a blocked terminal retransmits a packet during a slot will be a function of n which we denote f(n) so that

(22) $$g_i(n) = \binom{n}{i} [f(n)]^i [1-f(n)]^{n-i}$$

The following result can then be established :

<u>Theorem 4</u>.

A stationary retransmission control policy yields a stable broadcast channel if

$$\lambda = \sum_{i=1}^{\infty} ic_i < d$$

and an unstable one if $\lambda > d$ where $d = \lim_{n\to\infty}[c_1 g_0(n) + c_0 g_1(n)]$.

The proof of this result is given in Appendix 2. We do not have a proof of instability for $\lambda = d$ except for a special case; the question is only of mathematical interest, however.

Remark.

In fact Theorem 1 is a corollary of Theorem 4 since if (as is the case for the uncontrolled broadcast channel) f is independent of n we have d = 0.

Another consequence of Theorem 4 concerns the form which the function f(n) must take to ensure stability.

<u>Theorem 5</u>.

For the broadcast channel under stationary retransmission control to be stable it is necessary that

$$\lim_{n\to\infty} f(n) = 0 \quad \text{and} \quad \lim_{n\to\infty} nf(n) > 0$$

Proof.

Clearly if the first condition is not satisfied we shall have d = 0 leading to the instability of the channel. Now suppose that the second condition is not satisfied, that is $\lim n\to\infty \; nf(n) = 0$, but that the first condition is satisfied. Then $d = c_1$ and we cannot have $\lambda < d$; therefore by Theorem 4 the system will be unstable, which completes the proof.

We see therefore by this last result that a stationary retransmission control policy (with expected time between attempts of a blocked terminal to retransmit given by $[f(n)]^{-1}$) may stabilize the channel only if f(n) decreases with n but no faster then the function n^{-1}.

3.3. *An optimal retransmission control policy*.

It is natural to seek retransmission control policies which will maximize the output rate of the channel; for a stabilizing policy the maximum value will be d of theorem 4 since the input rate will be identical to the output rate. Consider

$$D_n(f) = c_1(1-f)^n + c_0 nf(1-f)^{n-1}$$

By deriving this expression with respect to f and setting the result equal to zero we see that $D_n(f)$ is maximized by setting f equal to $f^* = (c_0-c_1)(nc_0-c_1)^{-1}$ for $n \geq 1$, or $f^* = (1-\alpha)(n-\alpha)^{-1}$ if $\alpha = c_1/c_0$, where we are restricted to $\alpha < 1$ (for instance, with a Poisson arrival process $\alpha = \lambda$). The maximum value of $D_n(f)$ is then $D_n(f^*) = c_0[(n-1)(n-\alpha)^{-1}]^{n-1}$. In the limit as $n\to\infty$ we will obtain the throughput $d = \exp(\log c_0+\alpha-1)$.

If the arrival process is Poisson we obtain $d = e^{-1}$ as predicted by Abramson (1) and Kleinrock and Lam (8) for the maximum throughput of the channel.

On figure 3 we present time series characterizing channel beheviour obtained by Monte-Carlo simulation with a Poisson arrival process of packets from active terminals. On Figure 3a we show the behaviour of the uncontrolled broadcast channel; we see that if the number of blocked terminals is sufficiently high, the channel is unable to recover (i.e. it is unstable) and the total number of blocked terminals increases indefinitely while the channel throughput tends to zero. On Figure 3b we see the channel behaviour under identical conditions, except for the retransmission probability which is chosen to be f^*. The channel is now able to recover from an initial state with a large number of blocked terminals and the throughput matches the input rate. The exact form of f chosen in the simulation results of Figure 3b is $f^+ = (1-\lambda)n^{-1}$, where denominator term of f^* has been simplified. There is a simple intuitive (but non-rigorous) explanation for the choice of f^+ : when there are n blocked terminals and n is very large the set of blocked terminals will behave as a Poisson source of parameter $f^+.n = 1-\lambda$; thus the total input rate of packets to the channel will be $\lambda + f^+n = 1$ which is the maximum rate it can admit.

The control policy f^* could be approximately implemented by simple statistical estimation of the number of blocked terminals. The estimate could be obtained by a specialized terminal (or by the data concentrator which receives packets and sends back the acknowledgement packets) which would deduce an instantaneous estimate of the number of blocked terminals by measuring the throughput. It would then send once in a while an updated value of f^* on the frequency used for acknowledgement packets.

4. CONCLUSIONS.

In this paper we have given a theoretical treatment of some basic problems related to the packet switching broadcast channel. Its inherent instability has motivated us to look into stabilizing control policies. The first policy examined has been one in which access to the channel is controlled by admitting active terminals which wish to transmit a packet into an impeded set. Necessary and sufficient conditions under which the number of impeded plus blocked terminals remains bounded are derived, and it is shown that with this policy it is theoretically possible to achieve a throughput which is arbitrarily close to one.

We have then examined control schemes based only on choosing the transmission probability of any blocked terminal as a function of the total number of blocked terminals. Sufficient conditions for stability and instability of the channel and necessary conditions which must be satisfied by the retransmission probability are derived for this scheme. We then obtain the optimal control policy for the channel which maximizes the throughput. This policy appears promising as a practical means of optimizing channel performance.

Acknowledgements.

The authors are grateful to Prof. L. Kleinrock for discussions which motivated this research.

Appendix 1 : Proof of Theorem 3

The theorem is easily established for $\theta = 1$. It suffices to notice that in this case the system is equivalent to a single server queue with binomial service (with parameter $A = f$) and mean service time $1/f$; the arrival process is independent in each service interval. It is easily shown that the model has an equilibrium distribution of queue length (corresponding to the numbers of impeded terminals) if and only if $\lambda \leq A$. Now consider the case $\theta > 1$. Let π_n denote the equilibrium probability that the number of impeded terminals is n for $n \geq 1$; q_j will denote the equilibrium probability that there are zero impeded terminals and j blocked ones. The equilibrium probabilities satisfy for $n \geq 1$,

$$(23)\quad \pi_n = \pi_{n+1} C_0 A + \sum_{i=0}^{n-1} \pi_{n-i} [C_i(1-A) + C_{i+1}A] + \sum_{j=0}^{\theta} q_j C_{n-j+\theta} - q_\theta C_{n+1} A$$

Define the generating function $G(x) = \sum_{n=1}^{\infty} \pi_n x^n$. Then

$$(24)\quad G(x) = \frac{C_0 A}{x} (G(x) - \pi_1) + G(x)[G(x)(1-A) + \frac{A}{x} (\mathcal{C}(x) - C_0)] + \Phi(x) - q_\theta A \frac{\mathcal{C}(x) - C_0}{x}$$

where

$$\Phi(x) = \sum_{n=1}^{\infty} \sum_{j=0}^{\theta} q_j C_{n-j+\theta} x^n$$

yielding

$$(25)\quad G(x) = \frac{x \Phi(x) - q_\theta A(\mathcal{C}(x) - C_0) - C_0 A \pi_1}{x - x\mathcal{C}(x)(1 - A) - A\,\mathcal{C}(x)}$$

Notice that $\pi_n = p_{\theta+n}$, $n \geq 1$, $q_j = p_j$, $0 \leq j \leq \theta$, of equations (20), (21). By FOSTER's theorem [14] the Markov chain representing the number of blocked plus impeded terminals will be ergodic (and the channel will be stable) if there exists a positive solution to (23), of finite sum since the Markov chain is irreducible and aperiodic.

Suppose $q_0 > 0$; it can be easily shown that $q_j > 0$, $1 \leq j \leq \theta$, and $\pi_n > 0$, $n \geq 1$. Denote by $F(x)$ the numerator of (25) ; we first show that if $q_0 > 0$, then $F'(1) > 0$. We have

$$F'(1) = \Phi(1) + \Phi'(1) - q_\theta A \sum_{n-1}^{\infty} nC_n$$

$$= \sum_{n=1}^{\infty}\left(\sum_{j=0}^{\theta} q_j\, C_{n-j+\theta}\,(n+1) - Aq_\theta C_n n\right) > 0$$

since we have $A \leq 1$.

Now take $\lim_{x\to 1} G(x)$. After applying l'Hôpital's rule we remain with

$$\lim_{x\to 1} G(x) = \frac{F'(1)}{A - \mathcal{C}'(1)} = \frac{F'(1)}{A - \lambda}$$

Clearly if $A = \lambda$ and $q_0 > 0$ then $G(1)$ does not exist and the channel is unstable. Similarly if $A > \lambda$ and $q_C > 0$, then $G(1) < 0$ which is a contradiction so that again the channel is unstable. The case $\lambda < A$ however remains to be considered ; taking any finite $q_0 > 0$ we see that since $G(1) < \infty$ due to the fact that $F'(1)$ is bounded, the sum

$$G(1) + \sum_{j=0}^{\theta} q_j < \infty$$

and FOSTER's theorem is satisfied. Therefore if $\lambda < A$ the broadcast channel with the threshold control policy is stable.

Appendix 2 : Proof of Theorem 4

Let us first determine that the channel is unstable if $\lambda > d$. If the limit defining d exists, then for each $\epsilon > o$ there exists an integer n_o such that for all $n \geq n_o$,

$$|g_1(n) - a| \leq \epsilon \text{ and } |g_o(n) - b| \leq \epsilon$$

where a, b are constants such that

$$d = c_o a + c_1 b$$

Let $P(z) = \sum_{n=n_o}^{\infty} p_n z^n$, $Q(z) = \sum_{n=n_o}^{\infty} S_n z^n$. Then, from (9) and the discussion above we have

$$(26)\quad S_n - \sum_{j=o}^{n} S_{n-j} c_j \leq (a+\epsilon) p_{n+1} c_o + (b+\epsilon) p_n c_1$$

and

$$(27)\quad S_n - \sum_{j=o}^{n} S_{n-j} c_j \geq (a-\epsilon) p_{n+1} c_o + (b-\epsilon) p_n c_1$$

for all $n \geq n_o$. Thus, from (26), we derive

$$Q(z) - \sum_{n=n_o}^{\infty} \sum_{j=o}^{n} S_{n-j} c_j z^n \leq \frac{(a+\epsilon) c_o}{z} (P(z) - z^{n_o} p_{n_o} + (b+\epsilon) c_1 P(z))$$

Notice that

$$\sum_{n=n_o}^{\infty} \sum_{j=o}^{n} S_{n-j} c_j z^n = \sum_{n=n_o}^{\infty} \sum_{j=o}^{n-n_o} S_{n-j} c_j z^n + \sum_{n=n_o}^{\infty} \sum_{j=n-n_o+1}^{n} S_{n-j} c_j z^n$$

Therefore, if we denote $C(z) = \sum_{j=o}^{\infty} c_j z^j$,

$$(28)\quad Q(z)(1 - C(z)) \le \frac{(a+\epsilon)c_o}{z} (P(z) - z^{n_o} p_{n_o} + (b+\epsilon)c_1 P(z)) + \sum_{n=n_o}^{\infty} \sum_{j=n-n_o+1}^{n} S_{n-j} c_j z^n$$

The following relationship may be verified :

$$Q(z)(1 - z) = z^{n_o} S_{n_o} + P(z) - z^{n_o} p_{n_o} = P(z) + z^{n_o} S_{n_o - 1}$$

yielding after substitution in (28) and combining terms :

$$(29)\quad P(z) \left[\frac{1-C(z)}{1-z} - \frac{(a+\epsilon)c_o}{z} - (b+\epsilon)c_1\right] \le - z^{n_o}\left(\frac{1-C(z)}{1-z}\right) S_{n_o - 1} - z^{n_o - 1}(a+\epsilon)c_o p_{n_o} + \sum_{n=n_o}^{\infty} \sum_{j=n-n_o+1}^{n} S_{n-j} c_j z^n$$

However

$$\sum_{n=n_o}^{\infty} \sum_{j=n-n_o+1}^{n} S_{n-j} c_j z^n \le S_{n_o - 1} \sum_{n=n_o}^{\infty} \sum_{j=n-n_o+1}^{n} c_j z^n$$

and

$$\sum_{n=n_o}^{\infty} \sum_{j=n-n_o+1}^{n} c_j z^n \le \sum_{n=1}^{\infty} \sum_{j=n}^{\infty} c_j z^n = F(z)$$

where $\lim z \to 1 \; F(z) = \lambda$.

Returning to (29) we obtain :

$$(30) \quad P(z) \left[\frac{1-C(z)}{1-z} - \frac{(a+\epsilon)c_o}{z} - (b+\epsilon)c_1\right] \le S_{n_o-1}\left[-z^{n_o}\left(\frac{1-C(z)}{1-z}\right) + \sum_{n=1}^{\infty} \sum_{j=n}^{\infty} c_j z^n\right] - z^{n_o-1}(a+\epsilon)c_o p_{n_o}$$

Now take the limit as $z \to 1$ of both sides in (30). We obtain :

$$(31) \quad P(1)\,[\lambda - (a+\epsilon)c_o - (b+\epsilon)c_1] \le - c_o p_{n_o}(a+\epsilon)$$

Therefore if $\lambda > d$, choosing n_o sufficiently large so that $\lambda - d > \epsilon(c_o+c_1)$, we have that either $p_{n_o} = 0$ and $P(1) \le 0$ or $p_{n_o} > 0$ and $P(1) < 0$; in both cases it implies that the balance equations satisfied by the equilibrium probability distribution $\{p_n\}$ do not possess a positive solution. Thus the Markov chain representing the number of blocked terminals at the beginning of each slot is not ergodic and the channel is unstable if $\lambda > d$.

Starting with (27) and proceeding by arguments similar to the ones used above we can obtain

$$(32)\quad P(1)[\lambda - (a-\epsilon)c_o - (b-\epsilon)c_1] \geq -\lambda S_{n_o-1} - (a-\epsilon)c_o p_{n_o} + \sum_{n=n_o}^{\infty} \sum_{j=n-n_o+1}^{n} S_{n-j}c_j$$

The last term on the right-hand side of (32) cannot exceed λS_{n_o-1}, therefore assuming p_{n_o} is positive we may write

$$P(1)[\lambda - (a-\epsilon)c_o - (b-\epsilon)c_1] \geq -\alpha(n_o)$$

where

$$o < \alpha(n_o) = \lambda S_{n_o-1} + (a-\epsilon)c_o p_{n_o} - \sum_{n=n_o}^{\infty} \sum_{j=n-n_o+1}^{n} S_{n-j}c_j$$

since by choosing n_o sufficiently large we know that $a > \epsilon$. Therefore if $\lambda < d$, then

$$P(1) \leq \frac{\alpha(n_o)}{d-\lambda-\epsilon(c_o+c_1)}$$

assuming that n_o is large enough so that $d-\lambda > \epsilon(c_o+c_1)$. From (7) we notice that we may write for any $n \geq o$

$$(33)\quad p_n = k(n)p_o$$

where $k(n) > o$; thus

$$(34)\quad P(1) \leq \frac{\lambda \sum_{j=o}^{n_o-1} k(j) + (a-\epsilon)c_o k(n_o)}{d-\lambda-\epsilon(c_o+c_1)}$$

Notice that p_{n_o} is positive if and only if p_o is positive. We can now invoke FOSTER's theorem [14] which implies that the Markov chain is ergodic if there exists a positive solution to the equilibrium equation (7) such that $P(1) < \infty$. Setting $p_o=1$ (or any positive constant), (33) represents a positive solution of (7) ; by (34) we will have $P(1) < \infty$ therefore we have satisfied FOSTER's condition completing the proof that the channel is stable if $\lambda < d$. We now have to consider the case $\lambda = d$.

For $n \geq n_o$ we may write

$$g_1(n) = a + u_n \quad , \quad g_o(n) = b + v_n$$

so that from (9) we obtain

$$Q(z)\ [1-C(z)] = \frac{ac_o}{z}\ [P(z) - z^{n_o} p_{n_o}] + bc_1 P(z)$$

$$+ \frac{c_o}{z}\ [U(z) - z^{n_o} u_{n_o}\ p_{n_o}]$$

$$+ \quad c_1 V(z) + \sum_{n=n_o}^{\infty} \ \sum_{j=n-n_o+1}^{n} S_{n-j} c_j z^n$$

where

$$U(z) = \sum_{n=n_o}^{\infty} u_n p_n z^n \quad , \quad V(z) = \sum_{n=n_o}^{\infty} v_n p_n z^n$$

yielding

(35) $P(z) =$

$$\frac{-z^{n_o} S_{n_o-1} \frac{1-C(z)}{1-z} - C_o\,(a+u_{n_o}) p_{n_o} z^{n_o-1} + \frac{C_o}{z} U(z) + C_1 V(z) + \sum_{n=n_o}^{\infty} \sum_{j=n-n_o+1}^{n} S_{n-j} c_j z^n}{\frac{1-C(z)}{1-z} - \frac{aC_o}{z} - bc_1}$$

For $\lambda=d$, the denominator of P(1) vanishes. Instability for $\lambda=d$ will be verified if we can show that the numerator of P(1) does not, or that the numerator of P(z) tends to zero more slowly than the denominator for $\lambda=d$ as $z\to 1$. If $c_o g_1(n) + c_1 g_o(n) < d$ for all $n \geq n_o$ (i.e. if D_n tends to d from below), then $c_o U(1) + c_1 V(1) < 0$ and clearly the numerator of P(1) is negative for $p_{n_o} > 0$ and P(1) does not exist. Under this condition the channel is unstable for $\lambda=d$. In general however, even though we conjecture that the channel is unstable when $\lambda=d$, we have no proof of this.

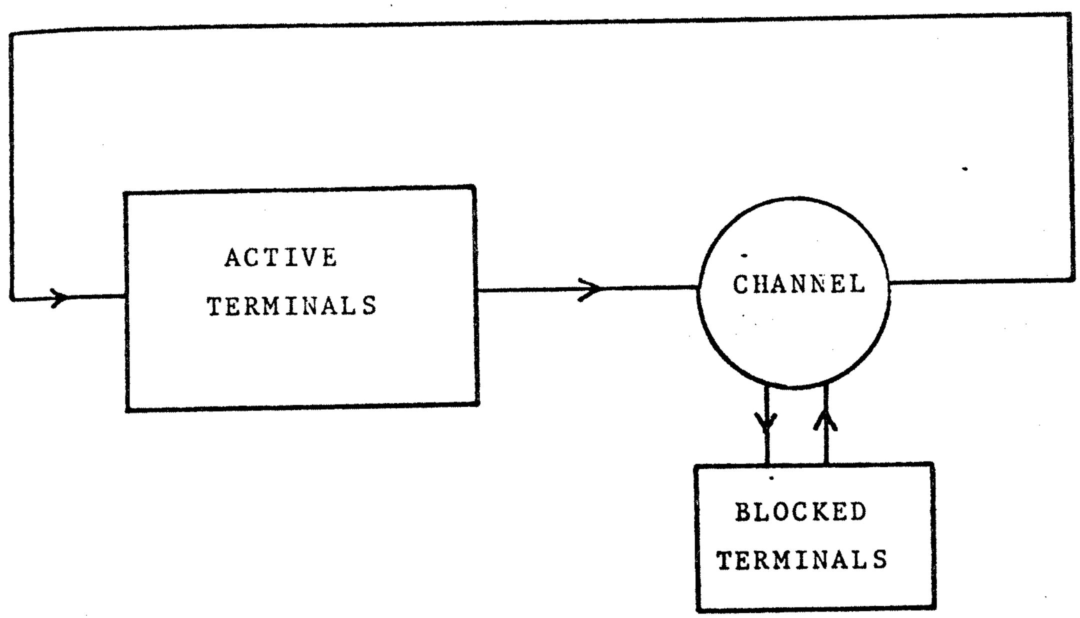


Figure 1

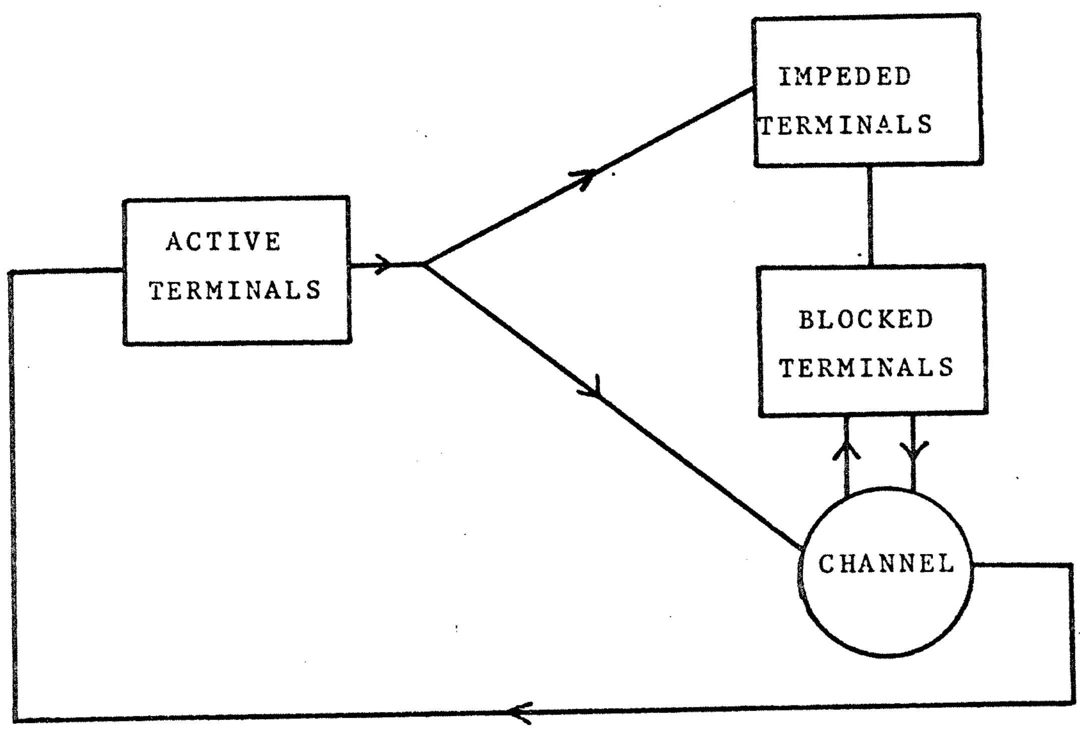


Figure 2

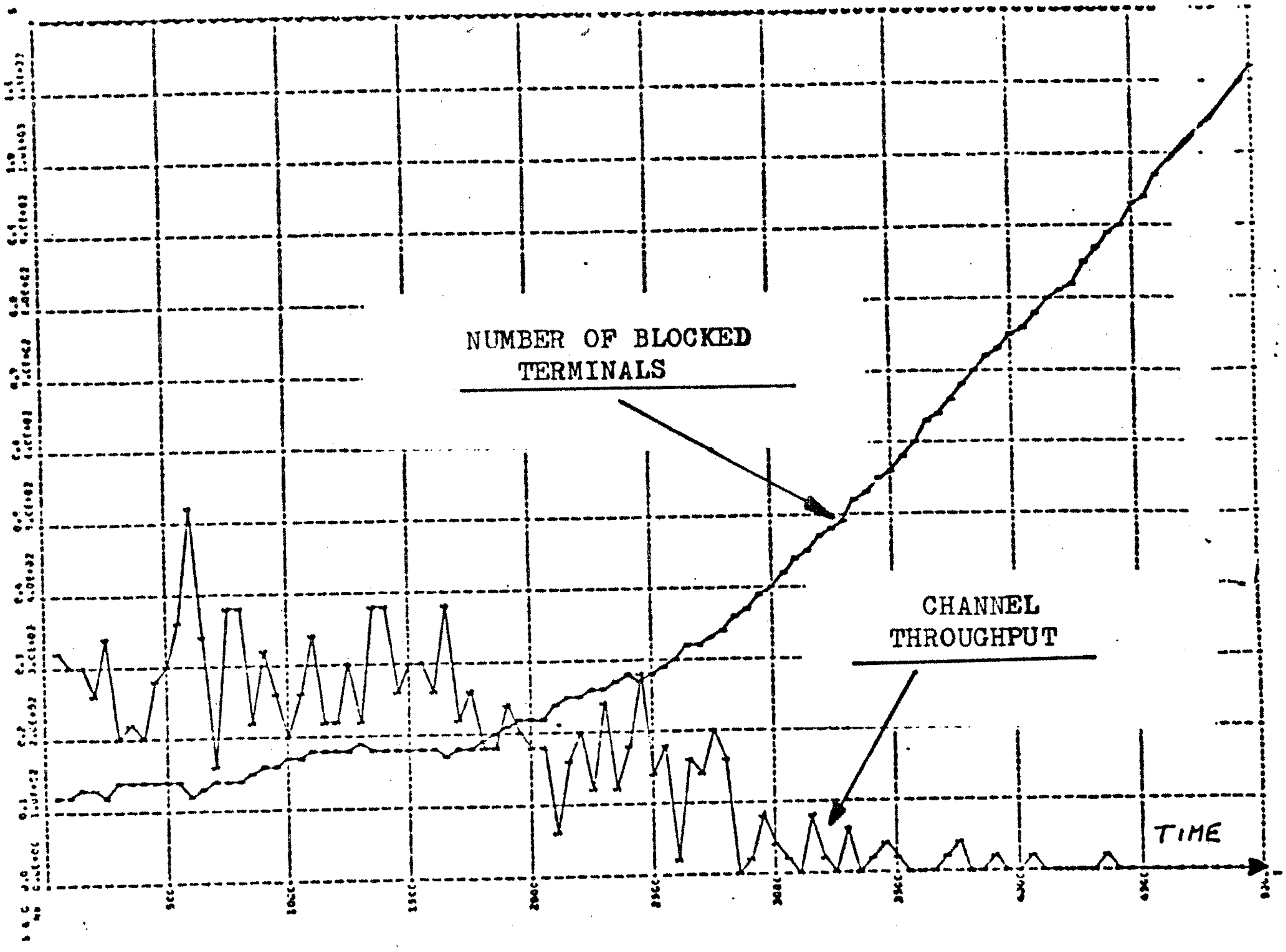
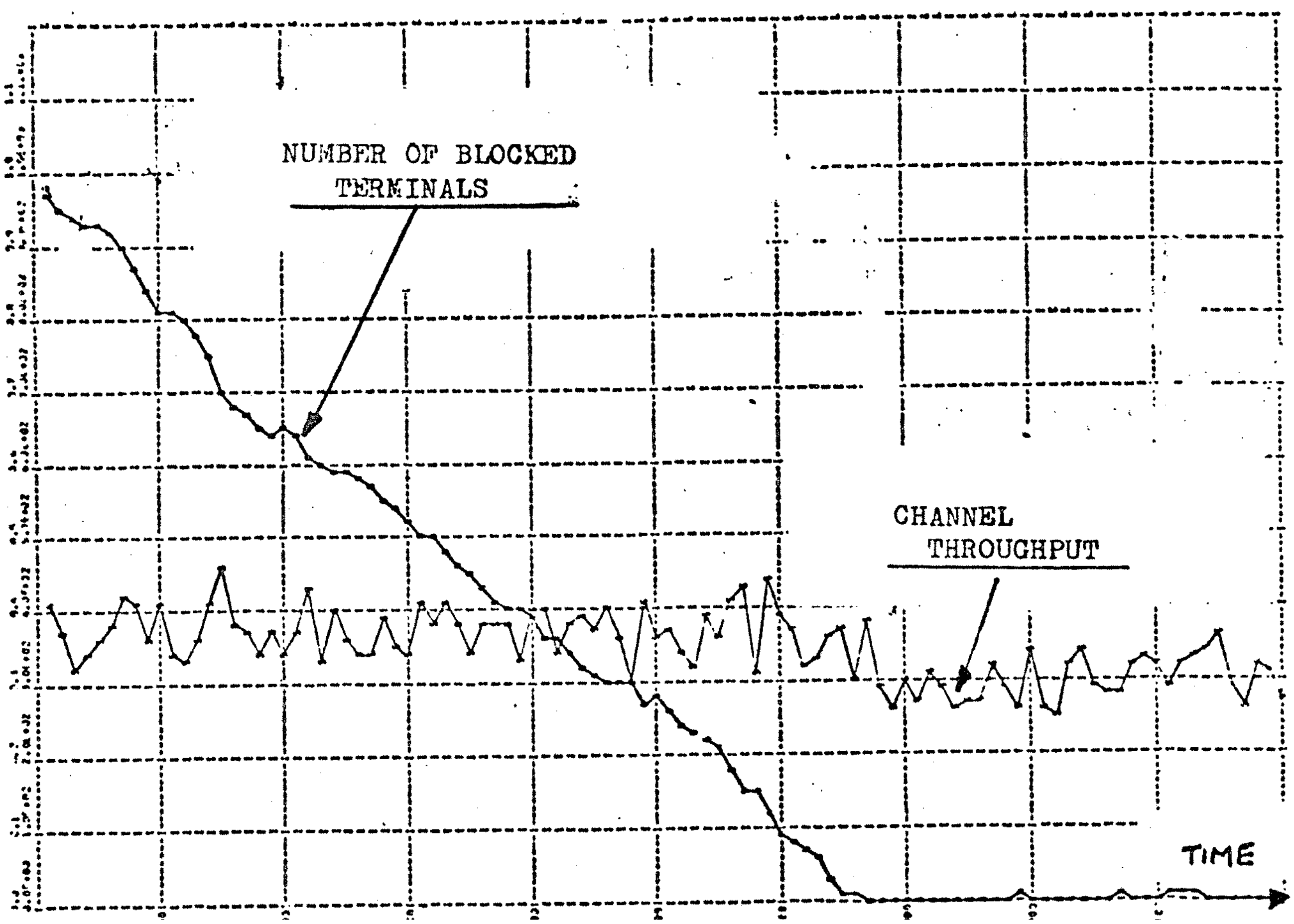

NUMBER OF BLOCKED TERMINALS
CHANNEL THROUGHPUT
TIME

Figure 3a


NUMBER OF BLOCKED TERMINALS
CHANNEL THROUGHPUT
TIME

REFERENCES

(1) Abramson, N., "The ALOHA system-Another alternative for computer communications", Fall Joint Computer Conference, AFIPS Conf. Proc., Vol. 37, 1970, pp. 281 - 285.

(2) Kleinrock, L., "Resource allocation in computer systems and computer-communication networks", Proc. IFIP Congress, Stockhom, 1974, to appear.

(3) Baran, P., Boehm, S., and Smith, P., "On distributed communications", series of eleven reports available from Rand Corp., Santa Monica, Calif., 1964.

(4) Davies, D. W., "The principles of a data communication network for computers and remote peripherals", Proc. IFIP Congress (Hardware), Edinburgh, 1968.

(5) Roberts, L. G., "Multiple computer networks and inter-computer communications", ACM Symposium on Operating Systems, Gatlinburg, Tenn., 1967.

(6) Abramson, N., "Packet switching with satellites", National Computer Conference, AFIPS Conf. Proc., 1973, PP. 695-702.

(7) Roberts, L. G., "Dynamic allocation of satellite capacity through packet reservation", National Computer Conference, AFIPS Conf. Proc., 1973, pp. 711-716.

(8) Kleinrock, L., and Lam, S., "Packet switching in a slotted satellite channel", National Computer Conference, AFIPS Conf. Proc., pp. 703-710.

(9) Kleinrock, L., and Lam, S., "On stability of packet switching in a random multi-access broadcast channel",

(10) Pouzin, L., "Presentation and major design aspects of the Cyclades computer network", submitted for presentation at the Third Data Communications Symposium, April 1973.

(11) Lam, S., Ph. D. Thesis, Computer Science Dept., UCLA, 1974.

(12) Fayolle, G., Gelenbe, E., Labetoulle, J., and Bastin, D., "The stability problem of broadcast packet switching computer networks", to appear in Acta Informatica.

(13) Howard. R.A., "Dynamic Probabilistic Systems,"Vol.II, John Wiley, New York, 1971.

(14) Cohen, J., "The Single Server Queue," North-Holland, Amsterdam, 1969.

A NON-MARKOVIAN DIFFUSION MODEL AND ITS APPLICATION TO THE APPROXIMATION OF QUEUEING SYSTEM BEHAVIOUR

E. Gelenbe

Résumé :

Par l'inclusion de temps d'arrêt sur les frontières distribuées selon des lois générales, nous étendons un modèle de diffusion au cas non-Markovien. Le modèle est appliqué au calcul des performances de modèles de systèmes informatiques.

Abstract :

We generalize a diffusion model of FELLER by the inclusion of holding times on the boundaries which are distributed according to a general probability distribution function. The stationary solution associated with the model is shown to depend only on the first moments of the holding time densities. The model is applied to the approximation of computer system behaviour.

1. Introduction

Models of multiple resource computer systems [2] have proved to be a good source of new problems and of general solution techniques [3-5] applicable to the analysis of networks of queues [1]. The research reported in this paper was motivated by this class of problems.

A promising method for the approximation of queueing systems with general service time distributions has originated with the work of GAVER [6] and NEWELL [7] who suggested the use of a diffusion process to approximate the number in queue. Consider for instance the GI/G/1 queue; basic to the diffusion approximation for this model is the assumption that as soon as a busy period begins (i.e. a customer arrives to a previously empty system) the stochastic process representing the number in queue is adequately approximated by the predictions of the central limit theorem which in reality are only valid asymptotically (as the duration of the busy period tends to infinity). Thus the behaviour of the number in the queue when it is non-empty is represented in [6], [7] by a probability density function $f(x,t)$ satisfying the diffusion equation

$$(1) \qquad -\frac{\partial}{\partial t} f(x,t) - b\frac{\partial}{\partial x} f(x,t) + \frac{\alpha}{2}\frac{\partial^2}{\partial x^2} f(x,t) = 0$$

where, $\{X(t), t \geq 0\}$ being the stochastic process approximating the number in queue,

$$(2) \qquad f(x,t)dx = \Pr\{x \leq X(t) < x+dx\} \quad .$$

Since the approach was intended for heavy traffic conditions it was also assumed that the lower boundary at $x = 0$ for the process $\{X(t), t \geq 0\}$ should act as a reflecting boundary. This last assumption implies that no probability mass can collect at $x = 0$, so that the model becomes inadequate in predicting the probability of an empty queue. GAVER and

SHEDLER [8] and KOBAYASHI [9], who generalized this approach to queueing networks, have chosen the integration constants in the solution to (1) with reflecting boundaries so that the model correctly predicts the stationary probability of an empty queue. In [10] some results predicted by the diffusion approximation are compared with measurements obtained from simulation experiments and with known mathematical results illustrating the excellent degree of accuracy obtainable by this new approach.

In a previous paper [11] we proposed the use of FELLER's elementary return process [12] instead of the diffusion process with reflecting boundaries to approximate a queueing system. This is a diffusion process with boundaries to which the process adheres for epochs which are exponentially distributed whenever the process attains a boundary ; at the end of the epoch the process is reinitialized according to a fixed probability density function. This model proved to be self contained and capable of predicting a number of results already known from queueing theory (such as the ergodicity condition or the probability of an empty queue in stationary state) exactly. In approximating a queue, the epoch during which the process adheres to the boundary at $x=0$ (queue length is zero) is meant to represent an idle period, and this period is not exponentially distributed in general. This motivated the non-Markovian diffusion model studied in this paper.

In Sections 2 and 3 we develop a diffusion model which generalizes the model of FELLER by the introduction of COXIAN density functions for the holding times of the process at the boundaries, i.e. density functions whose Laplace-Stieltjes transform is a rational function [13] . We then apply this model to the approximation of the GI/G/1 queue and to a computer system model in Sections 4 and 5. In Section 6 we treat the case where the holding times have arbitrary continuous and differentiable density functions and prove an interesting and useful theorem : it is shown that the stationary probability distribution in this model depends only on the first moments of these densities.

2. A Non-Markovian Diffusion Process

The diffusion process we shall present in this section appears to be new, though it is a generalization of standard diffusion processes [12]. To simplify and motivate the description of our model we shall imagine that the stochastic process $\{X(t), t \geq 0\}$, which will be used later to approximate the number in queue, represents the position of a particle moving on the closed interval $[0,M]$ of the real line. When the particle is in the open interval $]0,M[$ its motion is described by a Wiener process with drift; b and α, the instantaneous rate of change of the mean and variance of $X(t)$ in this region, are given by

$$b = \lim_{\Delta t \to 0} \frac{E\{X(t+\Delta t) - X(t)\}}{\Delta t} \tag{3}$$

$$\alpha = \lim_{\Delta t \to 0} \frac{E\{[X(t+\Delta t) - X(t)]^2\} - [E\{X(t+\Delta t) - X(t)\}]^2}{\Delta t} . \tag{4}$$

For our present purposes it is not necessary that b and α be functions of x, t; this restriction can be relaxed however.

When the particle reaches the lower boundary of the interval $[0,M]$ it remains there for a period of time h which is a random variable, at the end of which it "jumps" instantaneously back into the open interval $]0,M[$ to a random point whose position is defined by the probability density function $f_1(x)$. The probability distribution function of h is

$$F_h(r) = \Pr\{h \leq r\} \tag{5}$$

and is assumed to be a <u>COX or almost general distribution function</u> [13]. Since we shall be dealing with such distributions it may be useful to recall here their definition and main properties; we shall do this by visualizing what happens to the particle which attains the boundary $x = 0$. We shall imagine that the particle hitting the lower boundary and staying there for a random time h spends this time in traversing the network of Figure 1 which it enters at point A and leaves from B. This network is made up of a sequence of independent random delays each of which is exponentially distributed; the parameters of these exponential distribution functions are $\lambda_1,\ldots,\lambda_n$, respectively. After traversing the i-th delay the particle enters the $(i+1)$-th delay with fixed probability b_i or leaves the network of delays via B with probability $(1-b_i)$. Of course $b_n = 0$. The departure of the particle from the network of delays coincides with the end of h the holding time at the lower boundary and the instantaneous jump of the particle back into the open interval $]0,M[$. Let us denote by $f_h(r)$ the probability density function of h. From the previous description we have that $f_h^*(s)$ the Laplace transform of $f_h(r)$ is given by

$$(6)\qquad f_h^*(s) = \int_0^\infty e^{-sr} f_h(r)\,dr = \sum_{i=1}^{n} (1-b_i)a_i \prod_{j=1}^{i} \frac{\lambda_j}{(s+\lambda_j)}$$

where

$$(7)\qquad a_i = \begin{cases} 1 & \text{if } i = 1 \\ b_1\cdots b_{i-1} & \text{if } i > 1 \end{cases}$$

In fact (6) is the partial fraction expansion of the Laplace transform of a probability density function where the transform is a rational function of s if the b_i and λ_i, $1 \leq i \leq n$, are allowed to be complex quantities.[1)]Thus our <u>only</u> restriction on the random variable h is that the Laplace transform of its probability density function be a rational function. Such density functions have been introduced by COX [14]. Their interest lies in the fact that they can be used to approximate a density function by matching its first K moments, where K can be arbitrarily large. Also it is useful to note that there is a large body of literature on approximation by rational functions. Notice that the generality of (6) is limited by the fact that a positive probability that h is zero is not allowed; otherwise the expansion (6) would contain a constant term.

When the particle hits the upper boundary at $x = M$ it remains there for a random holding time H whose probability density function $f_H(r)$ is also almost general. The parameters in the network of delays corresponding to H, and similar to those for h shown in Figure 1, are $\mu_1,\ldots,\mu_m$ and $B_1,\ldots,B_m$ so that

$$(8) \qquad f_H^*(s) = \int_0^\infty e^{-sr} f_H(r)\,dr = \sum_{i=1}^{m} (1-B_i)A_i \prod_{j=1}^{i} \frac{\mu_j}{(s+\mu_j)}$$

where

$$(9) \qquad A_i = \begin{cases} 1 & \text{if } i = 1 \\ B_1\cdots B_{i-1} & \text{if } i > 1 \end{cases} .$$

[1)] As long as $f_h(r)$ is real.

At the end of the holding time at the upper boundary the particle jumps back instantaneously to a random point in]0,M[whose position is determined by the probability density function $f_2(x)$.

$f_1(x)$ and $f_2(x)$ may be taken to be functions of the instant at which the jumps occur. Notice that

(10) $$E\{h\} = \sum_{i=1}^{n} (1-b_i)a_i \sum_{j=1}^{i} \frac{1}{\lambda_j} .$$

This may be rewritten as

$$E\{h\} = \sum_{i=1}^{n} \frac{1}{\lambda_i} \sum_{j=i}^{n} (1-b_j)a_j$$

and it is easy to show that for $1 \leq i \leq n$,

(11) $$a_i = \sum_{j=i}^{n} (1-b_j)a_j .$$

Therefore

(12) $$E\{h\} = \sum_{i=1}^{n} \frac{a_i}{\lambda_i} .$$

Similarly

(13) $$E\{H\} = \sum_{i=1}^{m} \frac{A_i}{\mu_i} .$$

Let us introduce the notation

$$\text{(14)} \qquad \lambda = (E\{h\})^{-1}, \qquad \mu = (E\{H\})^{-1} .$$

If Var(h) denotes the variance of the random variable h, we see that some simple algebra yields

$$\text{(15)} \qquad \mathrm{Var}(h) = \sum_{i=1}^{n} \frac{a_i}{\lambda_i^2}, \qquad \mathrm{Var}(H) = \sum_{i=1}^{n} \frac{A_i}{\mu_i^2} .$$

That the process $\{X(t), t \geq 0\}$ defined in this section is non-Markovian is easy to see: once the particle is at any one of the boundaries the additional time it will remain there is not independent of the amount of time it has resided at the boundary up to the present instant. Let $\{k(t), t \geq 0\}$ be the stochastic process such that for all $t \geq 0$,

$$k(t) \in \{-n,-n+1,\ldots,0,1,\ldots,m\}$$

and

$$\text{(16)} \quad k(t) = \begin{cases} -j, \quad 1 \leq j \leq n, & \text{if the particle is in the } j\text{-th stage of the holding time } h \\ 0, & \text{if the particle is in }]0,M[\\ i, \quad 1 \leq i \leq m, & \text{if the particle is in the } i\text{-th stage of the holding time } H. \end{cases}$$

Then the process $\{(X(t),k(t)), t \geq 0\}$ has the Markov property.

3. The Equations for the Model

Let $f = f(x,t)$ denote the probability density function of the stochastic process $\{X(t),\ t \geq 0\}$ in the open interval $]0,M[$ and let $A_{x,t}$ and $C_{x,t}$ be operators defined by

$$(17) \qquad A_{x,t}f = -\frac{\partial}{\partial t} f - \frac{\partial}{\partial x} bf + \frac{1}{2}\frac{\partial^2}{\partial x^2} \alpha f$$

$$(18) \qquad C_{x,t}f = -bf + \frac{1}{2}\frac{\partial}{\partial x} \alpha f \quad .$$

Also let $P_i(t)$, $1 \leq i \leq n$, be the probability that the particle is in the i-th stage of the holding time at the lower boundary at time t while $Q_i(t)$, $1 \leq i \leq m$, is the probability that it is in the i-th stage of the holding time at the upper boundary at time t. The equations describing the evolution of the particle are

$$(19) \quad A_{x,t}f + \sum_{i=1}^{n} \lambda_i(1-b_i)P_i(t)f_1(x) + \sum_{i=1}^{m} \mu_i(1-B_i)Q_i(t)f_2(x) = 0$$

$$(20) \quad \frac{d}{dt}P_i(t) = \begin{cases} -\lambda_1 P_1(t) + C_{0,t}f , & \text{if } i = 1 \\ -\lambda_i P_i(t) + \lambda_{i-1}b_{i-1}P_{i-1}(t) , & \text{if } 1 < i \leq n . \end{cases}$$

$$(21) \quad \frac{d}{dt}Q_i(t) = \begin{cases} -\mu_1 Q_1(t) - C_{M,t}f , & \text{if } i = 1 \\ -\mu_i Q_i(t) + \mu_{i-1}B_{i-1}Q_{i-1}(t) , & \text{if } 1 < i \leq m . \end{cases}$$

where

$$C_{0,t}f = \lim_{x\to 0} \left[- bf + \frac{1}{2}\frac{\partial}{\partial x} \alpha f\right]$$

$$C_{M,t}f = \lim_{x\to M} \left[- bf + \frac{1}{2}\frac{\partial}{\partial x} \alpha f\right] \quad .$$

Define $P(t)$ the probability that the particle is at the lower boundary at time t, and let $Q(t)$ be the corresponding quantity for the upper boundary. We then have that

$$P(t) = \sum_{i=1}^{n} P_i(t) \ , \qquad Q(t) = \sum_{i=1}^{m} Q_i(t)$$

and

$$\text{(22)} \qquad \frac{d}{dt} P(t) = - \sum_{i=1}^{n} \lambda_i (1-b_i) P_i(t) + C_{0,t} f$$

$$\text{(23)} \qquad \frac{d}{dt} Q(t) = - \sum_{i=1}^{m} \mu_i (1-B_i) Q_i(t) - C_{M,t} f \quad .$$

Equations (19), (20), (21) are simple to interpret. Suppose Ω is a subinterval of $]0,M[$. Then (19) can be deduced from

$$\text{(24)} \qquad \frac{\partial}{\partial t} \int_{\Omega} f \, dx = \int_{\Omega} [- \frac{\partial}{\partial x} bf + \frac{1}{2} \frac{\partial^2}{\partial x^2} \alpha f] dx + \sum_{i=1}^{n} \lambda_i (1-b_i) P_i(t) \int_{\Omega} f_1(x) dx + \sum_{i=1}^{m} \mu_i (1-B_i) Q_i(t) \int_{\Omega} f_2(x) dx$$

which states that the rate of change of the probability mass in Ω is equal to the rate of flow of the probability mass via the boundaries of Ω (the first term on the right-hand-side of (24)) plus the rate of flow from $x = 0$ and from $x = M$ (the second and third terms, respectively, on the right-hand-side). In order to deduce (20), notice that for $1 < i \leq n$ we may write for any $t \geq 0$,

$$(25) \qquad P_i(t+\Delta t) = (1-\lambda_i \Delta t)P_i(t) + \lambda_{i-1} b_{i-1} \Delta t \; P_{i-1}(t)$$

since the time the particle spends in any one of the stages in Figure 1 is exponentially distributed; by collecting terms, dividing both sides by Δt and taking $\Delta t \to 0$, (25) yields (20) for $1 < i \leq n$ in the usual way. For obtaining (20) with $i = 1$ a similar procedure is applied if one notices that $C_{0,t} f$ is the flow of probability mass out of $]0,M[$ from the lower boundary and, of course, into the first stage of the holding time at $x = 0$. A similar interpretation can be given for (21); notice now that $-C_{M,t} f$ is the flow of probability mass away from $]0,M[$ via the upper boundary.

In addition to (19), (20), (21) appropriate boundary conditions for $f(x,t)$ must be specified and initial conditions (at $t = 0$) must be given for the stochastic process. Since the boundaries at $x = 0$ and $x = M$ behave as absorbing boundaries during their respective holding times we set $\lim_{x \to 0} f(x,t) = \lim_{M \to 0} f(x,t) = 0$ for all $t \geq 0$. Of course we set

$$\int_0^M f \, dx + P(t) + Q(t) = 1 \quad .$$

4. Application to the GI/G/1 Queue: Stationary Solution

In this section we propose an approximation to the number in a single server queue with general service time distribution of mean $1/\mu$ and variance V_s, independent of the interarrival times or of queue length, and with independent interarrival times having a general distribution function with mean $1/\lambda$ and variance V_a. The stochastic process

$\{X(t), t \geq 0\}$ approximating the number in queue at time t takes values on the non-negative real line $[0,\infty[$; it is of the type defined in the previous sections and its behavior is given by the equations

$$(26) \qquad A_{x,t}f + \sum_{i=1}^{n} \lambda_i(1-b_i)P_i(t)\delta(x-1) = 0$$

and

$$(27) \qquad \frac{d}{dt} P_i(t) = \begin{cases} -\lambda_1 P_1(t) + C_{0,t}f , & \text{if } i = 1 \\ -\lambda_i P_i(t) + \lambda_{i-1}b_{i-1}P_{i-1}(t) , & \text{if } 1 < i \leq n . \end{cases}$$

where $\delta(x)$ is the unit (probability) mass concentrated at $x = 0$. In this model the case $X(t) = 0$ refers to the empty queue; an arrival at time t to the empty queue corresponds to an instantaneous jump of $X(t)$ from 0 to 1, hence the value $f_1(x) = \delta(x-1)$. For a finite value of t there can be no probability mass at infinity hence we only have a probability mass $P(t)$ at the origin. When the queue becomes empty, the general arrival process is being approximated by an almost general process in the sense of Section 2 of this paper with appropriately chosen parameters $\lambda_1,\ldots,\lambda_n$ and $b_1,\ldots,b_n$. The parameters b and α in the operators $A_{x,t}$ and $C_{x,t}$ are chosen from the predictions of the central limit theorem as in [7,8,9]:

$$(28) \qquad b = \lambda - \mu$$

$$(29) \qquad \alpha = \lambda^3 V_a + \mu^3 V_s \quad .$$

λ, the mean arrival rate to the queue, coincides with $(E\{h\})^{-1}$, where

h is the random variable representing the holding time at the lower boundary for the diffusion process. Since there is no upper boundary to the process being considered the definition of μ given in (14) is irrelevant to the discussion in this section. Let us denote by $K_a = \lambda^2 V_a$ and $K_s = \mu^2 V_s$ the squared coefficients of variation of the interarrival and service times, respectively.

Let us denote by $f(x)$ and P_i, $1 \leq i \leq n$, the stationary distribution for $\{X(t), t \geq 0\}$; it is obtained (when it exists) by setting $\frac{\partial f}{\partial t} = 0$, $\frac{dP_i(t)}{dt} = 0$, $1 \leq i \leq n$, in the system of equations (26), (27). The appropriate boundary condition is still $\lim_{x \to 0} f(x) = 0$. P shall be the stationary total probability mass at the origin. The following result can be obtained.

Proposition 1. Let $\{X(t), t \geq 0\}$ be the stochastic process defined in this section approximating the number in queue for a GI/G/1 system. Its stationary probability distribution exists if $\rho = (\lambda/\mu) < 1$. Furthermore it is given by

$$f = \begin{cases} \rho[e^{-\gamma}-1]e^{\gamma x}, & \text{if } x \geq 1 \\ \rho[1-e^{\gamma x}], & \text{if } 0 \leq x \leq 1 . \end{cases} \tag{30}$$

$$P = 1 - \rho \tag{31}$$

$$P_i = \frac{\mu a_i}{\lambda_i} \rho(1-\rho) = \frac{\lambda a_i}{\lambda_i}(1-\rho) \tag{32}$$

where

(33) $$\gamma = -\frac{2(1-\rho)}{\rho K_a + K_s} \quad .$$

The approximate expected queue length at stationary state is given by

(34) $$\hat{L} = \int_0^\infty x f \, dx = \rho\left[\frac{1}{2} + \frac{\rho K_a + K_s}{2(1-\rho)}\right] \quad .$$

We shall not go through a proof of this result since it is very similar to that of Proposition 2 given in Section 5. It is interesting to notice that the diffusion approximation of this section is able to predict the stationary probability of an empty queue P exactly as one would expect to obtain from queueing theory; previous approaches to diffusion approximations [6,7,8,9] will not yield this result unless it is expressly introduced. Of course the fact that the total probability mass is unity

$$P + \int_0^\infty f \, dx = 1$$

has been used in obtaining our solution. Another interesting point is that $\hat{L}$ of (34) has a form similar to the POLLACZEK-KHINTCHINE formula for the M/G/1 queue which is

$$L = \rho\left[1 + \frac{\rho(1 + K_s)}{2(1-\rho)}\right] \quad .$$

In fact, if we set $K_a = 1$ in (34) in order to approximate a Poisson arrival process we obtain that the error is

$$L - \hat{L} = \frac{\rho}{2}[1 - K_s] \quad .$$

It is important to note that as $\rho \to 1$ the relative error $(L-\hat{L})/L$ vanishes, and that it vanishes as well when $K_s = 1$ (for the M/M/1 system). If we had discretized the probability density function $f(x)$ by defining the probability of i customers in queue as

$$\pi_i = \int_{i-1}^{i} f\, dx \, , \qquad \text{for } i \geq 1$$

and taken the average queue length as being

$$L' = \sum_{i=1}^{\infty} i\pi_i = \rho[1 + \frac{\rho K_a + K_s}{2(1-\rho)}]$$

we would have had

$$L - L' = -\frac{1}{2}\rho K_s \quad .$$

Again the relative error vanishes as $\rho \to 1$.

What is most interesting is that the stationary probability of an empty queue and all the moments of the stationary queue length distribution predicted by Proposition 1 are independent of all moments but the first of the holding time distribution at the origin.[1)]

It is essential to note that in our approximation method the random variable h refers to the time interval between the last departure from the queue of a busy period to the first arrival of the next busy period. If the arrival process is Poisson it is natural to take $\lambda = (E\{h\})^{-1}$,

---

1) This will be proved in greater generality in Section 6.

as we have done. However if the arrival process is not Poisson then the interarrival time distribution and the distribution of h need not be the same. Let us consider briefly the case where $(E\{h\})^{-1} = \lambda' \neq \lambda$. With this assumption the stationary solution of (26), (27) becomes

(35) $$f = \begin{cases} R[e^{-\gamma} - 1]e^{\gamma x}, & x \geq 1 \\ R[1 - e^{\gamma x}], & 0 \leq x \leq 1 \end{cases}$$

(36) $$P_i = \frac{\lambda' a_i}{\lambda_i}(1-R), \quad 1 \leq i \leq n$$

(37) $$P = 1 - R$$

where

(38) $$R = \frac{\lambda'}{\lambda' + \mu - \lambda} .$$

The condition for existence of the stationary solution is still $\rho = \lambda/\mu < 1$ since it is derived from the condition $\gamma < 1$. We see however that the usual queueing theory result $P = 1 - \rho$ will only be obtained if we set $\lambda' = \lambda$.

5. Application to a Closed Two-Server System With General Service Time Distributions

A special case of the model presented in Section 3 will be proposed here as an approximation to a queueing system containing a finite number of customers and two servers.

The system whose behaviour we wish to approximate is shown in Figure 2. It consists of a central processing unit (CPU) and an input-output device (IOD); a finite fixed number M of programs are being executed in the system. We shall assume in this section that service times at the CPU are independent and identically distributed (i.i.d.) random variables with distribution function with mean μ^{-1} and variance V_s; they are independent of the service times at the IOD which are also i.i.d. random variables of mean λ^{-1} and variance V_a. In general we do not exclude the possibility that λ, V_a, μ and V_s be functions of M.

The approximate model is described by the following set of equations, where the operators $A_{x,t}$ and $C_{x,t}$ are given by (17) and (18).

$$(39) \quad A_{x,t}f + \sum_{i=1}^{n} \lambda_i(1-b_i)P_i(t)\delta(x-1) + \sum_{i=1}^{m} \mu_i(1-B_i)Q_i(t)\delta(x-M+1) = 0$$

$$(40) \quad \frac{d}{dt} P_i(t) = \begin{cases} -\lambda_1 P_1(t) + C_{0,t}f , & \text{if } i = 1 \\ -\lambda_i P_i(t) + \lambda_{i-1}b_{i-1}P_{i-1}(t) , & \text{if } 1 < i \leq n \end{cases}$$

$$(41) \quad \frac{d}{dt} Q_i(t) = \begin{cases} -\mu_1 Q_1(t) - C_{M,t}f , & \text{if } i = 1 \\ -\mu_i Q_i(t) + \mu_{i-1}B_{i-1}Q_{i-1}(t) , & \text{if } 1 < i \leq m \end{cases}$$

Here $f = f(x,t)$ is the probability density function for $\{X(t), t \geq 0\}$, the process approximating the number of programs in the CPU queue, when this number is in $]0,M[$. The epoch from the instant that the CPU queue becomes empty to the arrival of the first customer is represented by the holding time h at the origin which is assumed to have an almost general distribution function (see Figure 1) of parameters $\lambda_1,\ldots,\lambda_n$ and $b_1,\ldots,b_n$. The corresponding parameters for H, the epoch from the instant that the IOD queue empties to the arrival of the first program to this queue, are $\mu_1,\ldots,\mu_m$ and $B_1,\ldots,B_m$. The boundary conditions are that $f(x,t)$ vanishes at $x = 0$ and at $x = M$. As in the previous section, we take $b = \lambda - \mu$ and $\alpha = \lambda^3 V_a + \mu^3 V_s$ as the parameters in the operators $A_{x,t}$ and $C_{x,t}$.

Proposition 2. Let $\{X(t),\ t \geq 0\}$ be the stochastic process approximating the number of programs in the CPU queue of the multiprogramming system described in this section, and whose probability distribution function satisfies (39), (40), (41) with the boundary conditions mentioned above. Its stationary probability density function exists and is given by

$$(42) \qquad f = \begin{cases} K[1 - e^{\gamma x}], & 0 < x \leq 1 \\ K[e^{-\gamma} - 1]e^{\gamma x}, & 1 \leq x \leq M-1 \\ K[e^{\gamma(x-M)} - 1]e^{\gamma(M-1)}, & M-1 \leq x < M \end{cases}$$

with P and Q, the probability masses at 0 and at M, respectively, at stationary state being

$$(43) \qquad P = K\frac{(1-\rho)}{\rho}, \quad Q = K(1-\rho)e^{\gamma(M-1)}$$

where $\rho = \lambda/\mu$, and

$$K = \rho(1 - \rho^2 e^{\gamma(M-1)})^{-1}$$

This result can be either verified by substitution in (39),(40),(41) with the appropriate boundary conditions and setting partial derivatives with respect to time equal to zero or can be obtained by solving the differential equations directly. The verification of the result is left to the interested reader. It is interesting once again to notice that the stationary solution obtained depends only on the expected values of the holding times at 0 and M.

On Figure 3 is summarized the result of some simulation experiments conducted by M. BADEL and V.Y. SHUM concerning our model predictions for the model described on Figure 2. The quantity plotted is the absolute value of the error term relative to the quantity obtained by simulation, for the stationary probability (1-P) that the CPU is active. That is, if $\eta =$ (1-P) obtained from the diffusion model and β is the corresponding quantity, then the quantity plotted is $|\eta - \beta|/\beta$. Two sets of simulation results, one with constant service time at the IOD and the other with exponentially distributed service time at the IOD are given. In each case we have also plotted the estimated confidence intervals for a 95°/$_o$ confidence level. The value of ρ has been varied between 0.25 and 0.9 and M has been varied between 1 and 10 ; the relative error plotted for each value of K_s is the maximum absolute relative error over all these values of ρ and M for a given value of K_s . This error remains relatively low, and in any case is smaller than the width of the confidence interval.

6. The Diffusion Model with General Holding Times at the Boundaries

Consider the following equations :

(44) $$\frac{dP(y,t)}{dt} = \frac{\partial P(y,t)}{\partial y} + \frac{\partial P(y,t)}{\partial t} = -u(y)\,P(y,t) + [C_{o,t}f]\,\delta(y)$$

(45) $$\frac{dQ(z,t)}{dt} = \frac{\partial Q(z,t)}{\partial z} + \frac{\partial Q(z,t)}{\partial t} = -v(z)\,Q(z,t) - [C_{M,t}f]\,\delta(z-M)$$

(46) $$A_{x,t}f + \left(\int_0^\infty u(y)\,P(y,t)\,dy\right) f_1(x) + \left(\int_0^\infty v(z)\,Q(z,t)\,dz\right) f_2(x) = 0$$

where $f(c, t) = f(M, t) = 0$, and

$$\int_0^M f(x,t)\,dx + \int_0^\infty P(y,t)\,dy + \int_0^\infty Q(z,t)\,dz = 1$$

(47) $$u(y) = \frac{g(y)}{1-G(y)}, \quad v(z) = \frac{h(z)}{1-H(z)}$$

$G(y)$, $H(z)$ are the probability distribution functions of total time spent by the process $\{X_t, t \geq 0\}$ on the lower and upper boundary, respectively, and $g(y)$, $h(z)$ are the corresponding densities (which are assumed to exist). The above system of equations generalizes (19), (20), (21) to the case where the holding time probability distribution functions $G(y)$, $H(z)$ are continuous but not necessarily Coxian. $f_1(x)$ and $f_2(x)$ are the density functions (definied on $[0, M]$) for the value taken by $\{X_t, t \geq 0\}$ at the time just after a jump from the lower and upper boundaries, respectively. The probabilities $P(y, t)$, $Q(z, t)$ are :

$$P(y,t) = Pr\,[X_\tau = 0,\ \tau \in\,]t-y, t] \text{ and } X_{t-y} > 0]$$

$$Q(z,t) = Pr\,[X_\tau = M,\ \tau \in\,]t-z, t] \text{ and } X_{t-z} < M]$$

The following result states that the stationary solution to (44), (45), (46) depends only on the expected values of the holding times at the upper and lower boundaries. Let

$$\lambda^{-1} = \int_0^\infty t\, g(t)\, dt, \quad \mu^{-1} = \int_0^\infty t\, h(t)\, dt$$

$$P(t) = \int_0^\infty dy\, P(y, t), \quad Q(t) = \int_0^\infty dz\, Q(z, t)$$

$P(t)$, $Q(t)$ are the probabilities that $X_t = 0$, $X_t = M$, respectively.

Theorem : Consider the stationary solution to (44), (45), (46) for $P(t)$, $Q(t)$, $f(x,t)$, obtained by setting

$$\frac{\partial P(y, t)}{\partial t} = 0, \quad \frac{\partial Q(z, t)}{\partial t} = 0, \quad \frac{\partial f}{\partial t} = 0$$

It depends on the expected values of the holding times at $x=0$, $x=M$, and is independent of all higher moments of the holding time distributions. Thus the stationary solution is identical to the corresponding solution when the holding times are exponentially distributed with means λ^{-1} , μ^{-1}.

Proof. : The stationary solution to (44) is $P(y) = \lim_{t\to\infty} P(y,t) = a(y)\exp(-\int_0^y u(s)ds)$ yielding $P(y) = a(y)(1-G(y))$, where $a(y)$ is obtained using the boundary condition $f(0,t)=0$ in (44) and carrying out the usual substitution , so that

$$a(y) = \left[\frac{\partial}{\partial x}(\alpha f)\right]_{x=0}$$

so that

$$P = \int_0^\infty P(y)\, dy = \frac{\lambda^{-1}}{2}\left[\frac{\partial}{\partial x}(\alpha f)\right]_{x=0}$$

similarly

$$Q = \frac{\mu^{-1}}{2}\left[-\frac{\partial}{\partial x}(\alpha f)\right]_{x=M}$$

also we obtain

$$\int_0^\infty u(y)\, P(y)\, dy = \frac{1}{2}\left[\frac{\partial}{\partial x}(\alpha f)\right]_{x=0} = \lambda P$$

$$\int_0^\infty v(z)\, Q(z)\, dz = \mu Q$$

so that (46) becomes at stationary state

$$(48) \qquad -\frac{\partial}{\partial x}(bf) + \frac{\partial^2}{\partial x}(\alpha f) + \lambda P f_1(x) + \mu Q f_2(x) = 0$$

which is independent of all moments but the first of $G(y)$, $H(z)$. This completes the proof.

This result is particulary interesting since it states that if we are only interested in the equilibrium solution to the non-Markovian diffusion model it suffices to solve the corresponding Markovian diffusion model.

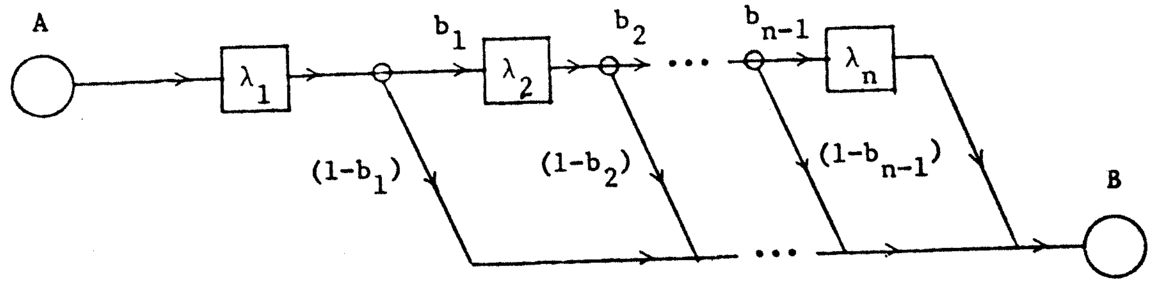


Figure 1

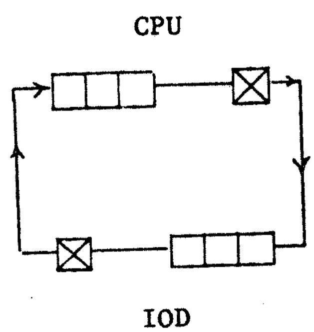


Figure 2

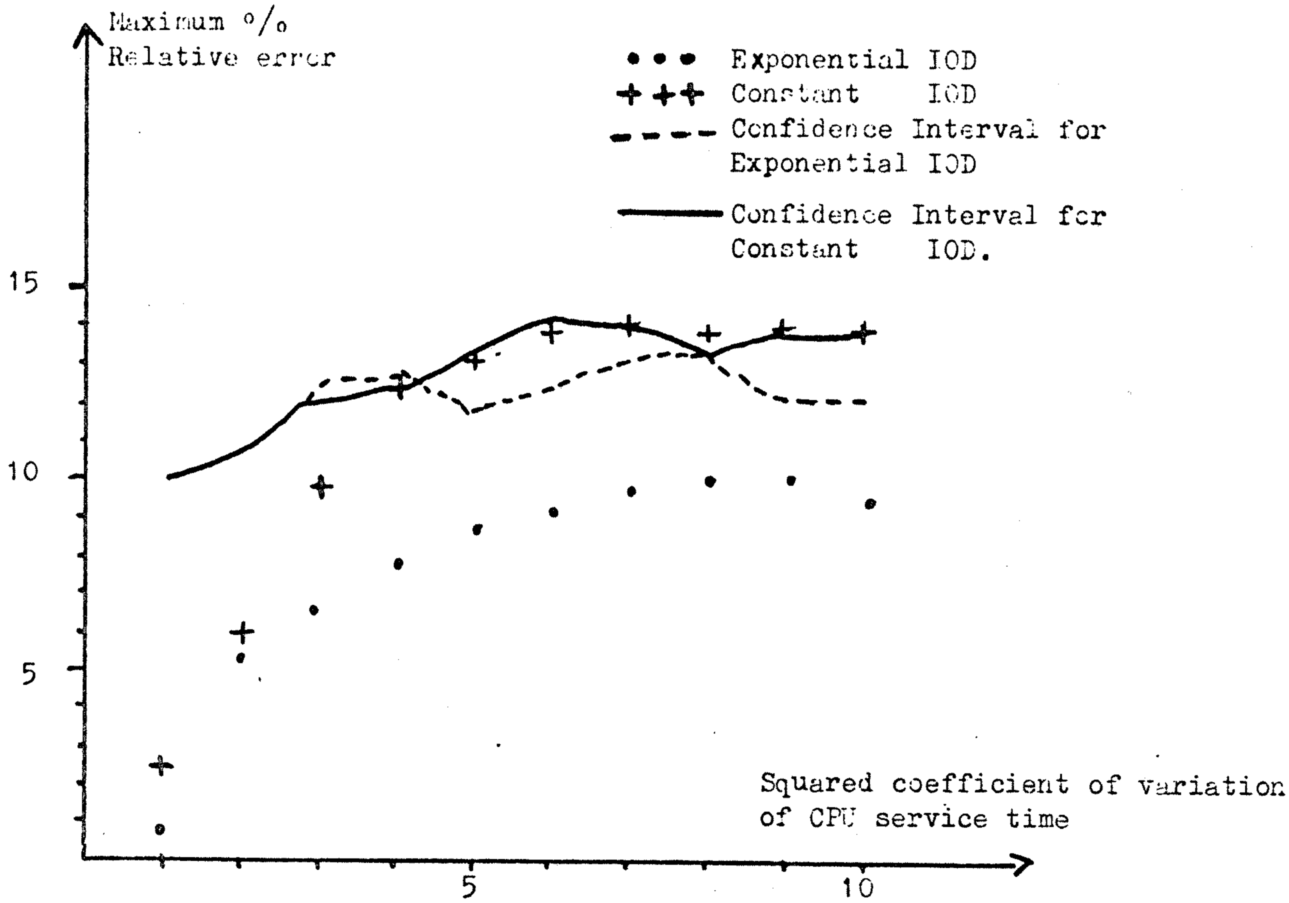

Maximum °/o
Relative error
Exponential IOD
Constant IOD
Confidence Interval for Exponential IOD
Confidence Interval for Constant IOD.
15
10
5
Squared coefficient of variation of CPU service time
5
10

Figure 3

REFERENCES

1. Jackson, J.R. "Jobshop-like queueing systems," Management Science 10, 131-142, 1963.

2. Moore, C.G. "Queueing network models of time-sharing systems," Ph.D. Thesis, Computer Science Dept., University of Michigan, Ann Arbor, 1970.

3. Muntz,R.R. "Poisson departure processes and queueing networks," IBM Research Report RC 4145, December 1972.

4. Chandy, K.M. "The analysis and solutions for general queueing networks," Proc. Sixth Annual Princeton Conf. on Information Sciences and Systems, Princeton University, Princeton, 1972.

5. Baskett, F., Chandy, K.M., Muntz, R.R., Palacios, J. "Open, closed, and mixed networks of queues with different classes of customers," JACM 22, 2, April 1975.

6. Gaver, D.P. "Diffusion approximations and models for certain congestion problems," J. Appl. Prob. 5, 607-623, 1968.

7. Newell, G.F. "Applications of queueing theory,"Chapman and Hall, London, 1971.

8. Gaver, D.P., Shedler, G.S. "Processor utilization in multiprogramming systems via diffusion approximations," Operations Research 21, 569-576, 1963.

9. Kobayashi, H. "Application of the diffusion approximation to queueing networks : Part I- Equilibrium queue distributions," JACM 21, 2, 316-328, April 1974.

10. Reiser, H., Kobayashi, H. "Accuracy of a diffusion approximation for some queueing networks," IBM J. Res. Dev. 18, 2, 110-124, March 1974.

11. Gelenbe, E. "On approximate computer system models," JACM 22, 2, 261-269, April 1975.

12. Feller, W. "Diffusion processes in one dimension," Trans. Am. Math. Soc., 77, 1-31, 1954.

13. Cox, D.R. "A use of complex probabilities in the theory of stochastic processes," Proc. Cambridge Philosophical Society, 51, 2, 313-319, April 1955.

14. Gelenbe, E., Muntz, R.R. "Probabilistic models of computer systems : Part I (Exact results)," to appear in Acta Informatica.

15. Gelenbe, E., Pujolle, G. "Probabilistic models of computer systems : Part II (Approximating the behaviour of a single queue in a network)," to appear in Acta Informatica.

16. Gelenbe, E. "A diffusion model for drum input-output operations with general interarrival distribution," in preparation.