NASA Technical Memorandum 80143

Application of Queueing Models to Multiprogrammed Computer Systems Operating in a Time-Critical Environment

Dave E. Eckhardt, Jr.

OCTOBER 1979

NASA

NASA Technical Memorandum 80143

Application of Queueing Models to Multiprogrammed Computer Systems Operating in a Time-Critical Environment

Dave E. Eckhardt, Jr.
Langley Research Center
Hampton, Virginia

National Aeronautics
and Space Administration

**Scientific and Technical
Information Branch**

1979

SUMMARY

A model of a central processor (CPU) which services background applications in the presence of time-critical activity is presented. The CPU is viewed as an M/M/1 queueing system subject to periodic interrupts by a deterministic, time-critical process. The Laplace transform of the distribution of service times for the background applications is developed. The use of state-of-the-art queueing models for studying the background processing capability of time-critical computer systems is discussed and the results of a model-validation study which support this application of queueing models are presented.

INTRODUCTION

A time-critical system is one in which the periodic and deterministic processing requirements of certain applications must be guaranteed in order that strict timing constraints can be met. The environment is said to be "hard real-time" since task deadlines are rigid. Unknown and untimely responses cannot be tolerated in such systems.

Real-time digital and hybrid simulations of physical systems (refs. 1 to 3) are examples of time-critical applications. Because the time-critical integrity of these applications must be maintained, a portion of system resources (central memory, I/O processors, channels, and disks) generally must be fully dedicated to their use. State-of-the-art computer systems, however, allow the remaining resources to be used by background batch and interactive applications (ref. 4). It is the performance of these systems when serving background applications that will be considered here.

Queueing theory has been used to model many aspects of computer systems, for example, batch and interactive systems, paged memory systems, I/O subsystems, communication networks, and systems with mixed classes of applications (refs. 5 to 8). The basis for applying this theory to multiprogrammed computer systems is the assumption that the population of user tasks, as a whole, behaves in a stochastic manner. For the normal multiprogrammed environment, this is a reasonable assumption. However, the deterministic nature of a time-critical process violates the Poisson assumptions of most queueing models, and it is not immediately evident that a queueing model can reasonably be applied to the study of systems which operate in this environment. This application of queueing models is the subject of this paper.

SYMBOLS

E_B event that server is busy

E_{BI} event that $\ell F \leqq T < F$

E_I	event that server is idle
E_{TI}	event that $0 \leq T < \ell F$
F	time between time-critical interrupts, or frame interval
ℓ	fraction of F required by time-critical task
ℓ_ϕ	$= \frac{\phi}{F}$
T	initiation time of virtual service
V	virtual service time random variable
W	system response time
α	$= e^{-\mu F(1-\ell)}$
λ	average arrival rate of background tasks
μ	CPU processing rate for background tasks
ρ	$= \frac{\lambda}{\mu}$
ρ'	$= \frac{\lambda}{\mu(1 - \ell)}$
ϕ	overhead time per frame

Notation:

$E(X)$	expected value of X
$f_X(x)$	probability density function (pdf) of random variable X
$f_X(x\|Y)$	pdf of X conditioned on Y
$X^*(s)$	Laplace transform of pdf of X

CPU MODEL

The CPU (fig. 1) is viewed as a single-server queueing facility in which a high-priority, time-critical task has immediate access to the CPU and interrupts the servicing of background tasks. The arrival of background tasks is a Poisson process with an average arrival rate λ. The required CPU service times for these tasks are assumed to be independent and identically distributed

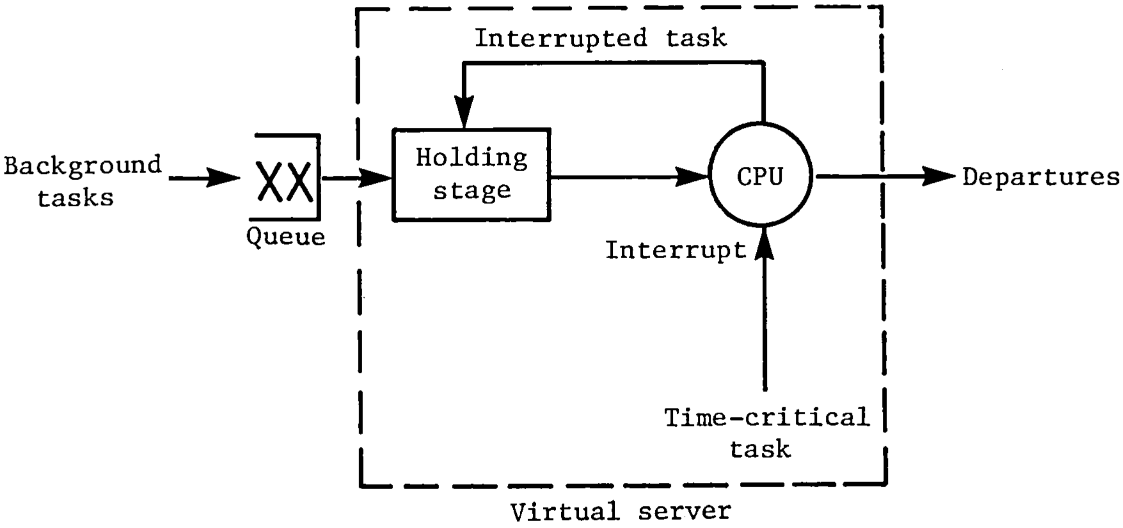

Figure 1.- Time-critical CPU model.

random variables from an exponential distribution with an expected service time μ^{-1}. The time-critical task is completely deterministic, unlike preemptive-priority queueing models or models in which the server is subject to breakdowns (refs. 9 and 10). This task arrives at a constant frame interval F and requires the CPU for a constant fraction ℓ of each frame. The time-critical process and the CPU server are combined to form a virtual server. The objective is to develop the resulting Laplace transform of virtual service time for the background tasks.

The virtual service time random variable V includes the required CPU service time of the background task and the accumulated holding time that occurs while the time-critical task is being serviced. The initiation of virtual service is independent of the time-critical process. That is, whenever both the CPU and holding stage are free of a background task, then an arriving background task can proceed to the holding stage immediately. Now if the time-critical task is processing, then the background task remains at the holding stage; otherwise, the background task receives CPU service immediately and no holding time is incurred.

During the first ℓF time units of each frame the time-critical task occupies the CPU, while for the remaining time $(1 - \ell)F$ the CPU is available for background processing. Since this process is identical for each frame, the beginning of a frame (the point at which the interrupt occurs) is used as a point of reference. The random variable T, measured from this reference point, is the time into the frame that a background task begins to receive CPU service.

It is easy to recognize the similarity between this model and a round-robin (RR) model, which includes an overhead penalty for exchanging tasks (for example, refs. 11 to 13). The RR system divides time into slices known as quantum intervals. Each task, in a cyclic fashion, is assigned a quantum interval and then exchanged for a new task at the end of the quantum. In practice some amount of overhead is required in each quantum to perform the task exchange. This overhead is equivalent to the time-critical service time ℓF, whereas the quantum interval is equivalent to the frame interval F. The primary difference is that, for the model considered here, virtual service can begin at any point in the frame, i.e., $0 \leqq T < F$, whereas for the RR model $T = 0$ when the overhead is assumed at the beginning of a quantum. Additionally, each task in the current model is completed before service on another background task begins.

Suppose that $T = 0$. If a background task is still processing at the end of the frame, then there is a probability α that additional service is required where

$$\alpha = \Pr[V > (1 - \ell)F] = e^{-\mu F(1-\ell)} \tag{1}$$

Since the probability that additional service is required is independent of the amount of service received during past frames (the memoryless property of the exponential), then the number of frames of service R is a random variable from the geometric probability density function (pdf),

$$\Pr[R=r] = \alpha^{r-1}(1 - \alpha) \qquad (r = 1, 2, 3, . . .) \tag{2}$$

During the frame in which a task departs from the system, the task will receive D units of CPU service, where D is from the truncated exponential pdf,

$$f_D(x) = \frac{\mu e^{-\mu x}}{1 - \alpha} \qquad (0 \leqq x < (1 - \ell)F) \tag{3}$$

which has a Laplace transform $D^*(s)$ given by

$$D^*(s) \overset{\Delta}{=} \int_{x=0}^{\infty} e^{-sx} f_D(x)\, dx = \frac{(1 - e^{-(s+\mu)(1-\ell)F})}{(s + \mu)(1 - \alpha)} \tag{4}$$

In general, however, T can occur in either of two intervals: the time-critical interval, defined as $0 \leqq T < \ell F$, or the background interval, defined as $\ell F \leqq T < F$. Consider first that virtual service begins during the background interval. Let E_{BI} represent this event. For a task beginning service in this interval there is an amount $F - T$ of CPU time available in the first frame. If this is a sufficient amount of processing time, then V has the normalized exponential pdf

$$f_V(x|E_{BI}) = \frac{\mu e^{-\mu x}}{1 - e^{-\mu(F-T)}} \qquad (0 \leqq x < F - T) \tag{5}$$

with the Laplace transform

$$V^*(s|E_{BI}, V \leqq F-T) = \frac{\mu(1 - e^{-(s-\mu)(F-T)})}{(s + \mu)(1 - e^{-\mu(F-T)})} \tag{6}$$

A task not completing service in this initial period will be interrupted by the time-critical process. In that case, the total virtual service time is the sum of the initial service $F - T$, an integral number of frames of interrupted service $(R - 1)F$, and the last frame of virtual service $\ell F + D$. Let

$$X = (F - T) + (\ell F + D) \tag{7}$$

the pdf of which has the Laplace transform

$$X^*(s) = e^{-s(F-T+\ell F)} D^*(s) \tag{8}$$

since $F - T + \ell F$ is constant for a given T. Let

$$Y = (R - 1)F \tag{9}$$

the pdf of which has the Laplace transform

$$Y^*(s) = \sum_{r=1}^{\infty} \alpha^{r-1}(1 - \alpha)e^{-s(r-1)F} = \frac{1 - \alpha}{1 - \alpha e^{-sF}} \tag{10}$$

Since X and Y are independent random variables, then

$$V^*(s|E_{BI}, V>F-T) = X^*(s)\ Y^*(s) = \frac{\mu(1 - e^{-(s+\mu)(1-\ell)F})(e^{-s(F-T+\ell F)})}{(s + \mu)(1 - \alpha e^{-sF})} \tag{11}$$

Using

$$\Pr\left[V>F-T\right] = e^{-\mu(F-T)} \tag{12}$$

and by the law of total probability,

$$\begin{aligned} V^*(s|E_{BI}) &= V^*(s|E_{BI},V>F-T)\ \Pr\left[V>F-T\right] + V^*(s|E_{BI},V\leq F-T)\left(1 - \Pr\left[V>F-T\right]\right) \\ &= \frac{\mu}{s+\mu}\left[1 + \frac{(e^{-s\ell F} - 1)\,(e^{-(s+\mu)(F-T)})}{(1 - \alpha e^{-sF})}\right] \end{aligned} \tag{13}$$

Suppose that virtual service begins during the time-critical interval. Let this event be E_{TI}. In that case there will be an initial delay of $\ell F - T$ before the background interval begins. The virtual service time for this condition is

$$V = \ell F - T + (R - 1)F + D \tag{14}$$

the pdf of which has the Laplace transform

$$V^*(s|E_{TI}) = \frac{\mu(1 - e^{-(s+\mu)(1-\ell)F})\,(e^{-s(\ell F-T)})}{(s + \mu)(1 - \alpha e^{-sF})} \tag{15}$$

Now consider the distribution of service time with regard to the number of background tasks that an arrival finds either queued or in virtual service. If an arrival finds no background tasks in the system, i.e., an idle virtual server, then the Laplace transform of virtual service will be either $V^*(s|E_{BI})$ or $V^*(s|E_{TI})$, depending on the point of arrival. Let E_I represent the idle virtual server condition. Removing the arrival-time condition, then

$$\begin{aligned} V^*(s|E_I) &= \int_{t=0}^{\ell F} V^*(s|E_{TI})\ f_T(t)\ dt + \int_{t=\ell F}^{F} V^*(s|E_{BI})\ f_T(t)\ dt \\ &= \frac{\mu(1 - \ell)}{s + \mu} + \frac{K_s}{F}\left[\frac{\mu^2}{s(s + \mu)^2}\right] \end{aligned} \tag{16}$$

where

$$K_S = \frac{(1 - \alpha e^{-sF(1-\ell)})(1 - e^{-sF\ell})}{1 - \alpha e^{-sF}} \tag{17}$$

and where

$$f_T(t) = \frac{1}{F} \tag{18}$$

because the distribution of interarrival times from a Poisson stream is known to be uniform. (See ref. 14, pp. 64 to 65, for a proof.) However, if the arrival finds a busy virtual server, denoted as the event E_B, then the virtual service of this arrival will begin at the departure point of the previous background task. Now since departures can only occur during the background interval, then an arrival to a busy server will always receive service from the distribution having a Laplace transform $V^*(s|E_{BI})$. Removing the service initiation-time condition results in

$$V^*(s|E_B) = \int_{t=\ell F}^{F} V^*(s|E_{BI})\ f_T(t|E_{BI})\ dt = \frac{\mu}{s + \mu} - \frac{K_S}{F}\left[\frac{\mu}{(1 - \ell)(s + \mu)^2}\right] \tag{19}$$

where

$$f_T(t|E_{BI}) = \frac{1}{(1 - \ell)F} \tag{20}$$

because interdeparture times of an exponential distribution are uniformly distributed and where K_S is given by equation (17).

Clearly then, the single-server queueing facility subjected to the interrupts of a time-critical process gives rise to a situation in which the first background arrival to initiate a busy period (having found an idle server) receives service from a distribution having a Laplace transform given by $V^*(s|E_I)$, whereas all other arrivals receive service from a distribution having a Laplace transform given by $V^*(s|E_B)$. Such a generalization of the M/G/1 queueing system has been studied by Welch (ref. 15). Welch shows that the expected system response time $E(W)$, which includes both queue wait time and service time, is given by

$$E(W) = E(V) + \frac{\lambda E(V^2)}{2\left[1 - \lambda E(V)\right]\left\{1 - \lambda\left[E(V|E_B) - E(V|E_I)\right]\right\}} \tag{21}$$

where $E(V)$ and $E(V^2)$ are the first and second moments of service time, respectively, and $E(V|E_B)$ and $E(V|E_I)$ are the expected values of the conditioned service times. Note that when $E(V|E_B) = E(V|E_I)$ the well-known Pollaczek-Khinchin formula for the M/G/1 queue is obtained. The Laplace transform of service time $V^*(s)$ is given by

$$V^*(s) = V^*(s|E_B)\,\Pr\left[E_B\right] + V^*(s|E_I)\left(1 - \Pr\left[E_B\right]\right) \tag{22}$$

where

$$\Pr\left[E_B\right] = \frac{\lambda E(V|E_I)}{1 - \lambda\left[E(V|E_B) - E(V|E_I)\right]} \tag{23}$$

is the probability of finding a busy server. Using the transforms given by equations (16) and (19), then

$$E(V|E_B) \overset{\Delta}{=} -\left.\frac{dV^*(s|E_B)}{ds}\right|_{s=0} = \frac{1}{\mu(1 - \ell)} \tag{24}$$

Similarly,

$$E(V|E_I) = \frac{1 + e_1}{\mu(1 - \ell)} \tag{25}$$

where

$$e_1 = \frac{\ell^2\left[\mu F(1 - \ell)(1 + \alpha) - 2(1 - \alpha)\right]}{2(1 - \alpha)} \tag{26}$$

Thus,

$$E(V) = \frac{1}{\mu(1 - \ell)} + \Delta E(V) \tag{27}$$

where

$$\Delta E(V) = \frac{e_1(1 - \rho')}{\mu(1 - \ell)(1 + \rho' e_1)} \tag{28}$$

and

$$\rho' = \frac{\lambda}{\mu(1 - \ell)} \tag{29}$$

The second moment of virtual service time is obtained as follows:

$$E(V^2|E_B) \overset{\Delta}{=} \left.\frac{d^2V^*(s|E_B)}{ds}\right|_{s=0} = \frac{2}{[\mu(1 - \ell)]^2} + \frac{2e_1}{\mu(1 - \ell)} \tag{30}$$

where e_1 is given by equation (26). Similarly,

$$E(V^2|E_I) = \frac{2}{[\mu(1 - \ell)]^2} + e_2 \tag{31}$$

where

$$e^2 = \frac{2\mu F\ell^2(1 - \ell)^2(1 + \alpha)}{1 - \alpha} + \frac{2(\mu F)^2\ell^2(1 - \ell)^2\alpha}{(1 - \alpha)^2} + \frac{(\mu F)^2\ell^3(1 - \ell)^2}{3}$$
$$+ 4\ell^3 - 6\ell^2 \tag{32}$$

Thus,

$$E(V^2) = E(V^2|E_B)\ \Pr[E_B] + E(V^2|E_I)\Big(1 - \Pr[E_B]\Big) = \frac{2}{[\mu(1 - \ell)]^2} + \Delta E(V^2) \tag{33}$$

where

$$\Delta E(V^2) = \frac{e_2 + \rho'(1 + e_1)(2e_1 - e_2)(1 + \rho' e_1)}{[\mu(1 - \ell)]^2} \tag{34}$$

Finally, from equation (21), the expected system response time is

$$E(W) = \frac{1}{\mu(1 - \ell)(1 - \rho')} + \Delta E(W) \tag{35}$$

where

$$\Delta E(W) = \Delta E(V) + \frac{\lambda\ \Delta E(V^2)}{2(1 - \rho')} \tag{36}$$

TIME-CRITICAL CPU ALLOCATION

The variables for this model are set by the requirements of the background workload and the time-critical process for a particular CPU and are presumably fixed. However, there is some flexibility in the allocation of CPU service to the time-critical task. In a practical sense, this allocation is constrained by the time in which results of computation must be available for transfer to output devices, i.e., the task deadline. It shall be assumed here that the deadline time is the same as the frame time, so that once an interrupt occurs there is no constraint on time-critical CPU allocation as long as ℓF units of processing time are realized by the process during each frame. We may consider, then, subdividing F into n equal quantum intervals of length $Q = F/n$ and in each of these quantums allocating ℓQ units of processing time for the time-critical process and the remaining time for background processing. Also assume now that a fixed overhead time ϕ is incurred during each quantum due to the alternation of service between the time-critical task and the background task. This overhead is expressed as some fractional portion ℓ_ϕ of the frame, i.e., $\ell_\phi = \phi/F$.

The effect of this action is that the time-critical load is more uniformly distributed over the frame, thus changing the characteristics of virtual service. The effects of this load-leveling action on the system response time can be calculated by replacing F with F/n and by adding the overhead $n\ell_\phi$, where $n\ell_\phi = \phi/Q$, to the required time-critical load ℓ. For example, suppose $\ell_\phi = 0.005$, $\ell = 0.500$, and $\rho = \lambda/\mu = 0.200$. The percentage improvement in response time due to load leveling, defined as

$$\text{Response-time improvement} = \frac{100\left[E(W|n=1) - E(W|n)\right]}{E(W|n=1)} \qquad (n = 1, 2, 3, \ldots) \qquad (37)$$

is shown in figure 2 for several values of the product μF.

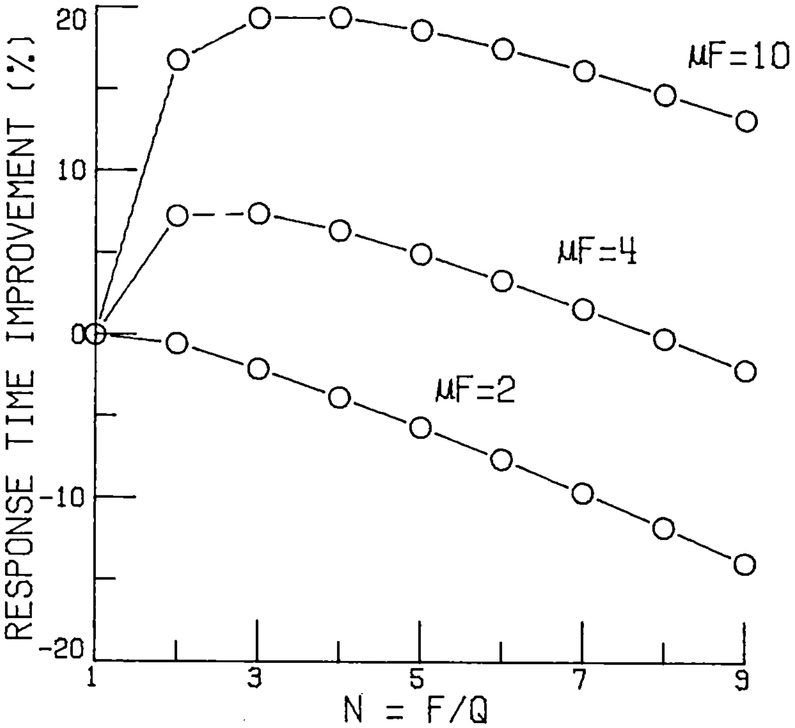

Figure 2.- Effect of time-critical load leveling on response time. $\ell = 0.5$; $\rho = 0.2$; $\ell_\phi = 0.005$. (N = n in the text.)

It is apparent that for smaller values of μF the overhead associated with task switching is the dominant factor and immediately degrades system performance. However, when the time-critical frame is large compared with the average

CPU service time for background tasks (large μF), then load leveling (to a point) would indeed improve system performance. This model should prove helpful in examining that proposition for a given system. Note that, for this particular set of parameters and for μF equal to 4 or 10, simply dividing F into two equal segments produces nearly optimal results.

APPLICABILITY OF STATE-OF-THE-ART QUEUEING MODELS

Consider now the feasibility of using existing queueing models to study time-critical systems. One advantage of this capability is the fact that software tools (refs. 7, 16, and 17) have been developed for these models which greatly simplify the process of model development and analysis. The model of Baskett, Chandy, Muntz, and Palacios (ref. 18) is a summarization and generalization of the results of many researchers and is representative of the state of the art in applying queueing theory to computer systems. This model allows one to distinguish between classes of applications, for example, batch and interactive, and can be used to study open (exogenous arrivals), closed (no exogenous arrivals), and mixed (closed with respect to some classes and open to other classes) networks of queueing facilities. Four different types of service facilities can be used to represent the various computer resources (CPU's, disks, terminals, etc.). The service distributions for these resources need only have rational Laplace transforms. If scheduling is first come, first served, however, the distribution must also be exponential. While this is a very general model, it does not provide a solution for preemptive-resume priority scheduling. Thus, if the time-critical process is modeled explicitly as a separate customer class, then this model is not appropriate, since preemptive-resume priority scheduling would be required. (Approximation techniques, however, have been developed for this scheduling discipline and have been used in some models, e.g., ref. 7.)

However, for the class of computer system considered here, it is proposed that it is not necessary to explicitly represent the time-critical task in order to study the performance of the background-processing system. As previously mentioned, because of stringent timing constraints, it is generally necessary to totally dedicate a portion of system resources to the time-critical applications. Such a system can be modeled as shown in figure 3. Conceptually the system is viewed as two machines: a background machine and a time-critical machine, each sharing the CPU but otherwise using mutually exclusive sets of resources. The performance of the time-critical machine is deterministic and the objective is to model the background machine.

The alternative to an explicit representation of the time-critical task is to implicitly include the effects of this task on the performance of the background machine by using the virtual service time distribution developed previously. However, when an exponential server is subjected to a time-critical load, the resulting service distribution is not exponential, nor is its Laplace transform (eq. (22)) rational. An approximation for this distribution is in order. Again, consider the magnitude of the product μF. A small product represents the average CPU service time for background tasks taking

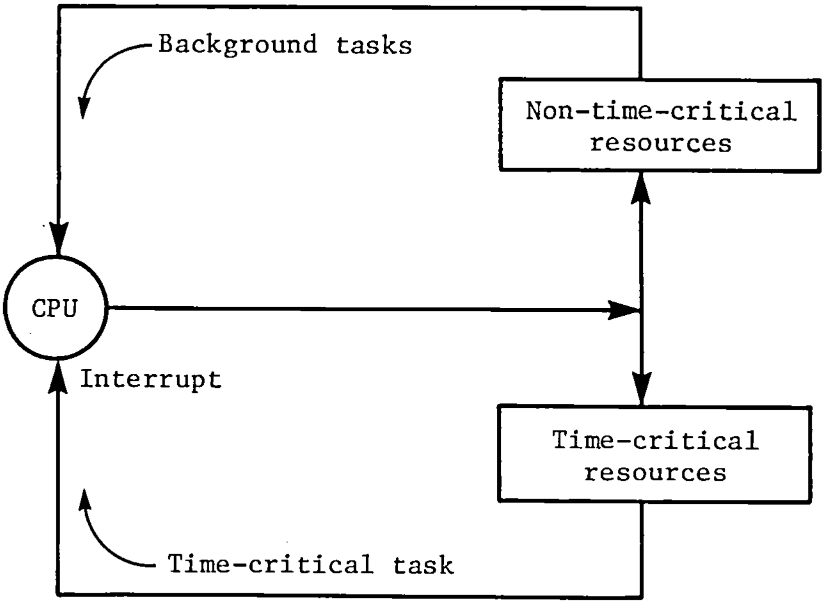

Figure 3.- A time-critical system with a background-processing capability.

place over many frames or, equivalently, a small F compared with the required CPU service time for background tasks. Referring to equations (26) and (32), it can be shown that

$$\lim_{\mu F \to 0} e_1 = 0$$

and

$$\lim_{\mu F \to 0} e_2 = 0$$

so that $\Delta E(V)$ in equation (28) and $\Delta E(V^2)$ in equation (33) will also approach zero as the frame becomes small compared with the required CPU service time. The result is that for small μF, an exponential server with parameter μ, operating under a time-critical load ℓ, appears to the input stream as an exponential server with parameter $\mu(1 - \ell)$. Thus, two independent processors emerge: a time-critical processor with a service rate of ℓ times the actual CPU service rate, and a background processor with a service rate of $1 - \ell$ times the actual CPU service rate. The total CPU utilization is the weighted utilization of each of these processors, that is,

$$\text{CPU utilization} = (1 - \ell)U_B + \ell U_{TC} \qquad (38)$$

where U_B and U_{TC} are the utilization of the background and time-critical processors, respectively. Note that if S is the service rate of the actual CPU (used a fraction ℓ of each frame by the time-critical task), then the time-critical processor with a service rate of $S\ell$ must be utilized 100 percent, so that $U_{TC} = 1$. Note also that the actual background CPU utilization is obtained by multiplying the background processor utilization U_B (obtained from a queueing model) by the factor $1 - \ell$.

This approximation is intuitively appealing and immediately offers a means for making use of existing queueing models. That is, one assumes an exponential server with the average service rate modified by the time-critical load. The accuracy of this approximation to the service distribution can be examined by forming the following:

$$\text{First moment error} = \frac{100\ \Delta E(V)}{1/[\mu(1 - \ell)]} \tag{39}$$

$$\text{Second moment error} = \frac{100\ \Delta E(V^2)}{2/[\mu(1 - \ell)]^2} \tag{40}$$

These are shown first as a function of ℓ and ρ' for $\mu F = 1$ in figures 4 and 5. It is found that the maximum errors occur at the midregion of ℓ. Then, for $\ell = 0.5$, the above errors are shown in figures 6 and 7 as a function of μF for several values of ρ'. These latter figures, then, show the maximum errors that can be expected for a given value of the product μF as a result of the approximate service distribution. Clearly, the magnitude of the product μF is a good indicator of the accuracy of the approximation.

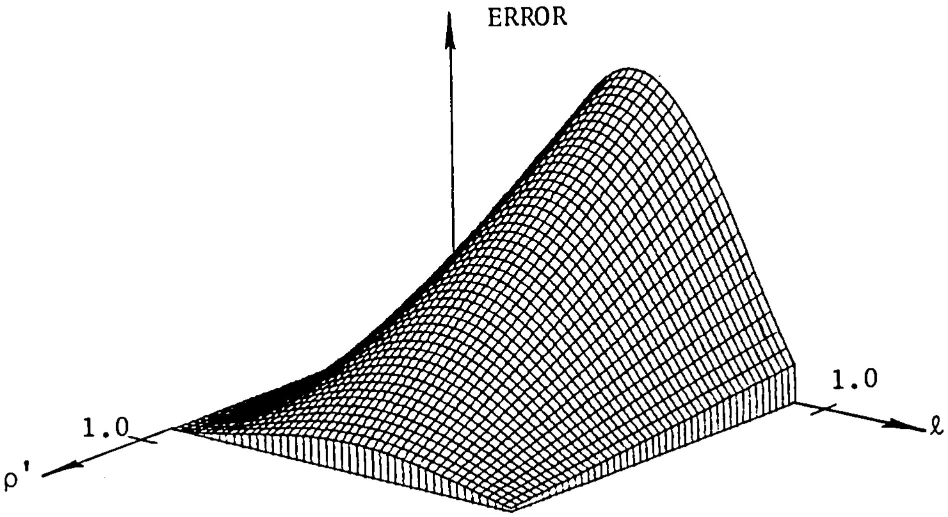

Figure 4.- First moment error. $\mu F = 1$.

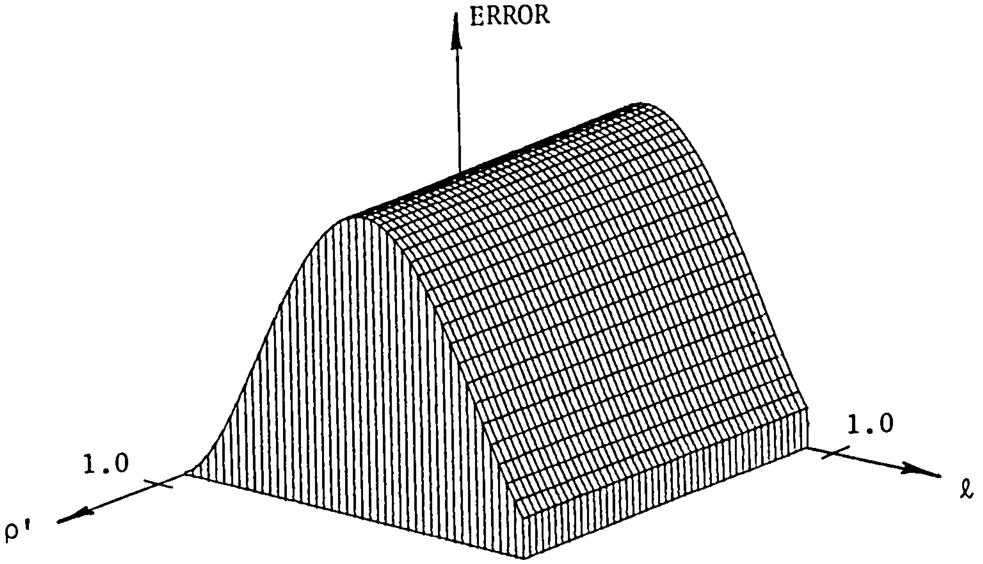

Figure 5.- Second moment error. μF = 1.

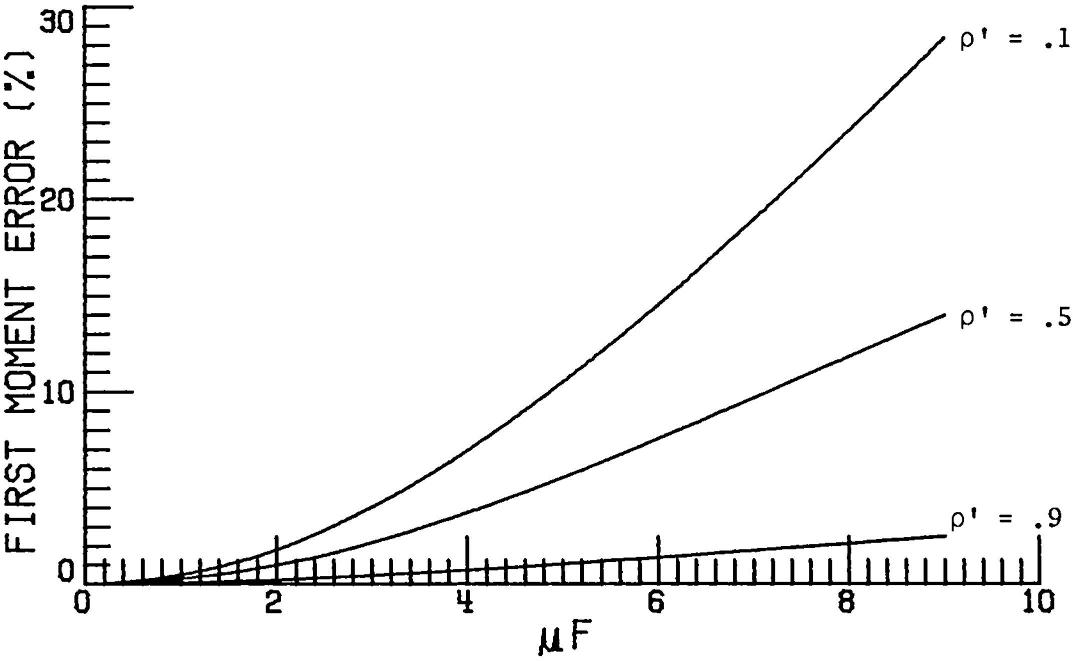

Figure 6.- First moment error. ℓ = 0.5.

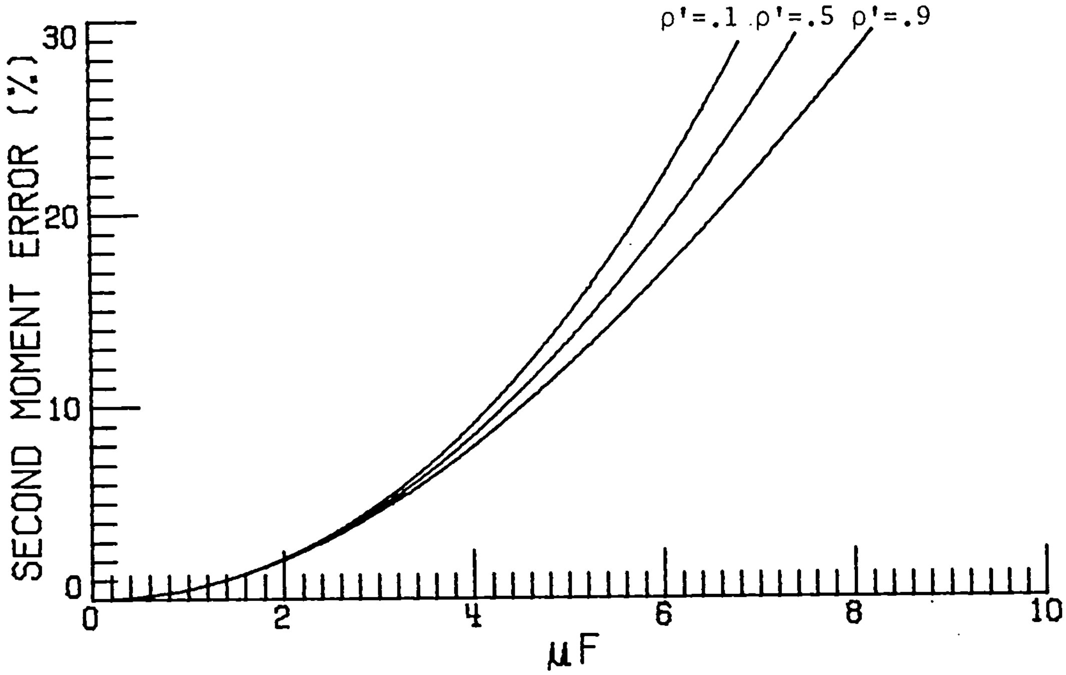

Figure 7.- Second moment error. ℓ = 0.5.

EXPERIMENTAL RESULTS

The results of the previous section were applied to a performance evaluation study of a digital simulation facility located at the NASA Langley Research Center in Hampton, Virginia (ref. 19). This facility, is capable of supporting batch, interactive, and time-critical applications. However, in order to run controlled experiments, only the batch and time-critical capabilities were used. System measurements determined that the average batch CPU service time was 12.20 msec. Because the majority of simulation applications used a frame time of 31.25 msec, the resulting μF of 2.6 indicated that the average CPU service rate for a time-critical load ℓ could reasonably be formulated as

$$\mu(\ell) = 12.2^{-1}(1 - \ell) \tag{41}$$

A closed queueing model was developed for the batch processing system. Controlled experiments were made in a purely batch environment in order to calibrate and validate the model. Then a time-critical task was introduced into the system and the experiments were repeated. The CPU service rate of equation (41) was used and degree of multiprogramming of the model (average number of circulating batch tasks) was modified to account for the dedicated use of central memory by the time-critical task. A comparison of measured results and model results for CPU utilization and CPU queue length is shown in the following table. As can be seen, once a calibrated batch model was obtained, the effects of the time-critical task were accurately predicted by the model.

Time-critical CPU utilization ℓ, %	Batch CPU utilization, %		Average CPU queue length	
	$(1 - \ell) \times$ model	Measured	Model	Measured
0 (Calibration)	60.9	60.5	1.5	1.3
10.0	59.7	59.3	1.7	1.6
30.0	54.4	53.8	2.6	2.2
50.0	44.7	45.0	3.9	3.5
70.0	29.5	29.3	5.8	5.5

CONCLUDING REMARKS

The Laplace transform of the service distribution for an M/M/1 queueing system subject to the periodic interrupts of a deterministic time-critical process has been developed. By using this distribution, the benefits of uniformly distributing the load of the time-critical process can be evaluated and weighed against the increased overhead that is incurred due to the alternation of service between the time-critical task and the background task.

The Laplace transform of this distribution does not have a rational form, so state-of-the-art network queueing models are not directly applicable. However, if the time between interrupts is small compared with the average CPU service time of the background tasks, then an approximate service distribution is an exponential distribution, with the original service rate multiplied by the fraction of processing time available to background tasks. This suggests that for an appropriate range of model parameters, current queueing models can reasonably be extended to the study of background processing in time-critical environments. Experimental results are reported which support this application.

Langley Research Center
National Aeronautics and Space Administration
Hampton, VA 23665
September 14, 1979

REFERENCES

1. Belluardo, R.; Gocht, R.; and Paquette, G.: A Time-Shared Hybrid Simulation Facility. AFIPS Conference Proceedings, Volume 28 - 1966 Spring Joint Computer Conference, Spartan Books, Inc., c.1966, pp. 355-363.

2. Fineberg, Mark S.; and Serlin, Omri: Multiprogramming for Hybrid Computation. AFIPS Conference Proceedings, Volume 31 - 1967 Fall Joint Computer Conference, Thompson Book Co., c.1967, pp. 1-13.

3. Gracon, T. J.; Nolby, R. A.; and Sansom, F. J.: A High Performance Computing System for Time Critical Applications. AFIPS Conference Proceedings, Volume 39 - 1971 Fall Joint Computer Conference, AFIPS Press, c.1971, pp. 549-560.

4. Gracon, Thomas J.: Large Scale Simulation Systems: The Evolution of a State of the Art System. Summer Computer Simulation Conference, June 1972, pp. 352-364.

5. Buzen, J. P.: Queueing Network Models of Multiprogramming. M.S. Thesis, Harvard Univ., 1971.

6. Moore, Charles G., III: Network Models for Large-Scale Time-Sharing Systems. Tech. Rep. No. 71-1 (Contract N00014-67-A-0181-036(NR 049-311)), Univ. of Michigan, Apr. 30, 1971. (Available from DDC as AD 727 206.)

7. Buzen, Jeffrey P.: A Queueing Network Model of MVS. ACM Comput. Surv., vol. 10, no. 3, Sept. 1978, pp. 319-331.

8. Wong, J. W.: Queueing Network Modeling of Computer Communication Networks. ACM Comput. Surv., vol. 10, no. 3, Sept. 1978, pp. 343-351.

9. Avi-Itzhak, B.; and Naor, P.: Some Queueing Problems With the Service Station Subject to Breakdown. Oper. Res., vol, 11, no. 3, May-June 1963, pp. 303-320.

10. White, Harrison; and Christie, Lee S.: Queueing With Preemptive Priorities or With Breakdown. Oper. Res., vol. 6, no. 1, Jan.-Feb. 1958, pp. 79-95.

11. Adiri, I.; and Avi-Itzhak, B.: A Time-Sharing Queue With a Finite Number of Customers. J. Assoc. Comput. Mach., vol. 16, no. 2, Apr. 1969, pp. 315-323.

12. Rasch, Philip J.: A Queueing Theory Study of Round-Robin Scheduling of Time-Shared Computer Systems. J. Assoc. Comput. Mach., vol. 17, no. 1, Jan. 1970, pp. 131-145.

13. Kleinrock, Leonard: Swap-Time Considerations in Time-Shared Systems. IEEE Trans. Comput., vol. C-19, no. 6, June 1970, pp. 534-540.

14. Kleinrock, Leonard: Queueing Systems. Volume II: Computer Applications. John Wiley & Sons, Inc., c.1976.

15. Welch, Peter D.: On a Generalized M/G/1 Queueing Process in Which the First Customer of Each Busy Period Receives Exceptional Service. Oper. Res., vol. 12, no. 5, Sept.-Oct. 1964, pp. 736-752.

16. Keller, Tom W.: User's Manual for the ASQ (Analytic Solution of Queues) System. Information Res. Assoc., c.1973 (rev. Sept. 1976).

17. Reiser, M.: Interactive Modeling of Computer Systems. IBM Syst. J., vol. 15, no. 4, 1976, pp. 309-327.

18. Baskett, Forest; Chandy, K. Mani; Muntz, Richard R.; and Palacios, Fernando G.: Open, Closed, and Mixed Networks of Queues With Different Classes of Customers. J. Assoc. Comput. Mach., vol. 22, no. 2, Apr. 1975, pp. 248-260.

19. Eckhardt, Dave E., Jr.: Modeling and Performance Evaulation of Multiprogrammed, Time-Critical Computer Systems. D. Sc. Diss., George Washington Univ., 1979.

1. Report No. NASA TM-80143	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle
APPLICATION OF QUEUEING MODELS TO MULTIPROGRAMMED COMPUTER SYSTEMS OPERATING IN A TIME-CRITICAL ENVIRONMENT

5. Report Date
October 1979

6. Performing Organization Code

7. Author(s)
Dave E. Eckhardt, Jr.

8. Performing Organization Report No.
L-13209

9. Performing Organization Name and Address
NASA Langley Research Center
Hampton, VA 23665

10. Work Unit No.
505-31-83-02

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, DC 20546

13. Type of Report and Period Covered
Technical Memorandum

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A model of a central processor (CPU) which services background applications in the presence of time-critical activity is presented. The CPU is viewed as an M/M/1 queueing system subject to periodic interrupts by a deterministic, time-critical process. The Laplace transform of the distribution of service times for the background applications is developed. The use of state-of-the-art queueing models for studying the background processing capability of time-critical computer systems is discussed and the results of a model-validation study which support this application of queueing models are presented.

17. Key Words (Suggested by Author(s))
Systems analysis
Computer-systems modeling
Queueing models

18. Distribution Statement
Unclassified - Unlimited

Subject Category 66

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	19	$4.00

* For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1979