International Journal of Computer and Information Sciences, Vol. 7, No. 2, 1978

Multivariate Point Process Models for Response Times in Multiprogrammed Systems

David W. Hunter[1] and Gerald S. Shedler[2]

Received April 1976

We consider the formulation of marked multivariate point process models for job response times in multiprogrammed computer systems. Complementing queueing network representation of the structure of the system to be modeled, the particular R-process (*R*esponse time process) model we propose permits representation of resource contention, facilitates the incorporation of realistic workload characteristics into system performance predictions, and can reproduce inhomogeneities observed in running systems. Specification of the structure of the R-process model is conditional on workload marks; this effectively separates the difficult problem of formal representation of workload characteristics from the overall problem of response time prediction. To illustrate these ideas, an application to database management systems is considered. Evidence of the predictive capability of the R-process model, based on statistical analysis of response time data from an IMS system, is also given.

KEY WORDS: Performance models; multiprogrammed systems; response times; marked multivariate point processes; statistical analysis of trace data; database management systems.

1. INTRODUCTION

This paper is concerned with the stochastic modeling of multiprogrammed computer systems, with emphasis on the prediction of job response times. Motivated by the desire to facilitate the incorporation of realistic workload

[1] IBM T. J. Watson Research Center, Yorktown Heights, New York.
[2] IBM Research Laboratory, San Jose, California.

0091-7036/78/0600-0193$05.00/0

characteristics into system models from which useful performance predictions can be efficiently obtained, we consider the construction of marked multivariate point process models. Complementing widely used queueing network representations of the interactions between the processing and input–output resources of multiprogrammed systems, such point processes model resource contention among job streams and should be useful in prospective performance evaluation studies.

Networks of queues provide a convenient means of representing the interaction between the processing and input–output resources of (multiprogrammed) computer systems and subsystems. In general, the representation of the structure of a system as a network of queues, at a level of detail deemed appropriate, can be readily accomplished. Formal representation of workload characteristics of the system within a network of queues, however, is more difficult. One of the advantages of a queueing network representation is that the flow of jobs through the system is made explicit and can be easily visualized by means of one or more graphical displays associated with the network. Also, the interpretation of service centers in the network (e.g., hardware components of the system), along with service times and job routing, is usually apparent, and selection of parameters for the system representation is facilitated by these interpretations.

The setting down of stochastic fine structure assumptions (e.g., arrival processes, service time distributions, and routing probabilities) within such a queueing network system representation gives rise to a queueing network model. There is much literature dealing with such models.[3,6,13,14,16] Under the usual convenient, but not necessarily realistic, queueing-theoretic assumptions (e.g., independent and identically, often exponentially, distributed service times) analyses of queueing network models based on a "numbers-in-queue" state space can be carried out (see refs. 1, 7, 11, and 17), yielding expressions for stationary queue length distributions that can be evaluated numerically. Measures of system performance derivable from the stationary queue length distributions such as device "utilizations" and job "throughput" can then be obtained from such analyses.

Other measures of system performance (calculated as sums of queueing times) involve the distribution of times for a job to traverse a portion of the network. Certain such times (in closed networks complete circuits or loops, and in open networks times from source to sink) are often interpretable as job *response times*; these response times are likely to be particularly sensitive to workload characteristics. Analyses based on the numbers-in-queue state space yield expected values for response times, but do not yield other characteristics of interest such as percentiles or quantiles. Since alternative analyses to provide these characteristics are in general not available, it is necessary to undertake simulation studies of the queueing network. For

certain classes of closed and finite capacity open queueing networks, efficient regenerative process methods have recently been developed for estimation of general characteristics of long-run response times through simulation.[8,9] For simulation of response times in more complex queueing networks used in detailed system simulations, little information seems to be available. It is our feeling that such simulations are inherently difficult and likely to be time-consuming and costly to carry out.

In this paper we consider the formulation of marked multivariate point process (MMPP) models for job response times in multiprogrammed systems. A (univariate) stochastic point process provides a formal probabilistic structure for a series of events occurring in time. In a multivariate point process, there are events of two or more types.[5] The events in an MMPP, in addition to having a type, carry a real, possibly vector-valued mark. In our context, event types generally will provide qualitative information about the multiprogrammed processing of jobs (e.g., job start, job termination, and job stream identity) whereas event marks will provide quantitative workload information.

We envision that prospective point process models of this kind can be developed from a queueing network system representation, and that the inputs required will be similar to those required for a queueing network model. We anticipate, however, that the point process models will facilitate the incorporation of realistic workload characteristics into performance studies and be capable of reproducing the gross time inhomogeneities (nonstationary behavior) observed in running systems.[15] Such inhomogeneities (e.g., in job response times and throughput) generally result from changing workload characteristics. Although important for understanding system performance, these inhomogeneities are often not reflected in performance models. We also anticipate that these point process models can be simulated more efficiently to obtain characteristics of job response times than can comparable queueing network models.

In Sec. 2, some examples are given which lead to consideration of a particular multivariate point process model (having limited dependence and restrictions on event types). A formal definition of this model, termed an *R*-process (*R*esponse-time process), appears in Sec. 3. The selection and interpretation of parameters in an *R*-process model based on a queueing network system representation is discussed in Sec. 4. Evidence of the predictive capability of the *R*-process model is given in Sec. 5, where results are reported of an analysis of response time data obtained from a database management system. Parameter estimation procedures are given along with a comparison of the data with *R*-process simulation output. Some remarks and directions for further work are contained in a final section.

2. DESCRIPTION OF THE MODEL AND EXAMPLES

The particular marked multivariate point processes we consider, R-processes, are built up from a finite number of (dependent) marginal point processes. Let J be a positive integer and consider $2J$ (mutually exclusive) *event types*, denoted $a_1, a_2, \ldots, a_J, b_1, b_2, \ldots, b_J$. With each j, $1 \leqslant j \leqslant J$, associate a *bivariate point process* (series of events of two types) composed of alternating event types a_j and b_j. Each event within a marginal bivariate point process carries a real, possibly vector-valued *mark*. The marks are such that an event of type a_j and the immediately following event of type b_j are identically marked. The *total series of events* comprising the R-process is obtained by superposition of the J marginal processes. For $n \geqslant 1$, denote by $S_n^{(j)}$ the time of the nth event of type a_j after an arbitrarily chosen time origin. Similarly, denote by $T_n^{(j)}$ the time of the nth event of type b_j after the first type a_j event following the time origin. Also, define $T_0^{(j)}$ to be the time of the first event of type b_j after the origin, provided that it occurs before time $S_1^{(j)}$; otherwise define $T_0^{(j)}$ equal to 0. In terms of these $\{S_n^{(j)}\}$ and $\{T_n^{(j)}\}$, define times between events in the jth marginal process by

$$R_n^{(j)} = T_n^{(j)} - S_n^{(j)}, \qquad n \geqslant 1$$

and

$$Q_n^{(j)} = S_{n+1}^{(j)} - T_n^{(j)}, \qquad n \geqslant 0$$

Finally, denote the mark on the event at time $S_n^{(j)}$ by $W_n^{(j)}$. See Fig. 1 for a graphical presentation of these quantities. Some motivation is provided by the following examples.

Example 1. Consider the closed queueing network of Fig. 2 (see ref. 8). There are a fixed number J of jobs denoted $1, 2, \ldots, J$ in the network. Upon completion of service in center 1 which renders α service, in accordance with a binary-valued variable ψ, the job joins the tail of the queue in center 1 (when $\psi = 1$) or (when $\psi = 0$) joins the tail of the queue in center 2 which renders β service. Neither center 1 nor center 2 service is subject to interruption. Both queues are assumed to be serviced according to a first-in-first-out (FIFO) discipline. A quantity of interest is R, which is the time required by a particular job to enter the tail of the center 2 queue after completing a center 2 service. To associate a marked multivariate point process with the queueing network of Fig. 2, view an event of type a_j as occurring when job j completes service at center 2 and joins the tail of the queue in center 1. Similarly, view an event of type b_j as occurring when job j joins the tail of the queue in center 2. Identify with $W_n^{(j)}$ the number of center 1 services received by job j between its $(n-1)$st and nth center 2 services. Then $R_n^{(j)}$ is the nth occurrence of the time R for job j.

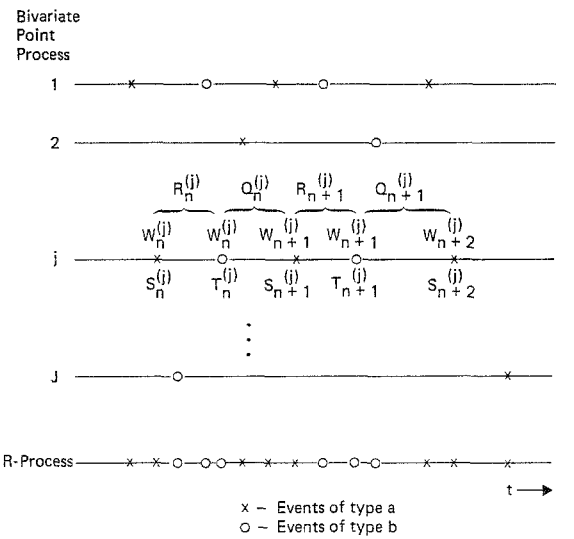

Fig. 1. R-process as superposition of bivariate point processes.

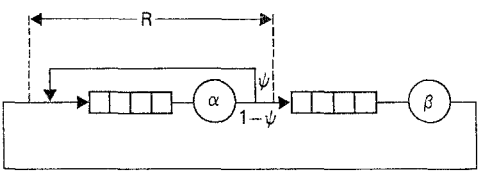

Fig. 2. Closed queueing network.

Example 2. Consider the finite capacity open queueing network of Fig. 3 (see ref. 12). Jobs arrive to the network from an external source and depart to an external sink. Feedback to the queue in center 1 occurs in accordance with the binary-valued variable ψ. The waiting room in center 2 is assumed to be finite. The center 1 server is blocked and ceases to render service when there are K jobs waiting or in service at center 2. Service at the first center resumes when the queue length in center 2 falls to $K-1$. Jobs arriving when the network already contains J ($\geqslant K$) jobs are turned away.

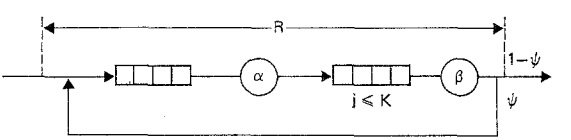

Fig. 3. Finite capacity open queueing network.

Both queues are assumed to be serviced according to a FIFO discipline. A quantity of interest is the total time spent by a job in the system, indicated by R in Fig. 3.

Associate a marked multivariate point process with the network of Fig. 3 by assigning a number $1 \leqslant j \leqslant J$ to an arriving job as follows. Assign to an arriving job the smallest possible number, subject to the restriction that no two jobs in the system may have the same number. View an event of type a_j as occurring when a job arrives and is assigned the number j; similarly view an event of type b_j as occurring when a job numbered j departs from the system. Identify with $W_n^{(j)}$ the number of times the nth job numbered j undergoes feedback to center 1 upon completion of service at center 2. Then $R_n^{(j)}$ is the total time spent in the system by the nth job numbered j.

Example 3. Consider the closed queueing network of Fig. 4. The interpretation of this diagram differs from that of a conventional queueing network in that services are distinguished from the servers that perform them. Thus circles in the figure represent services rather than servers. Five types of service, denoted by α_0, α_1, α_2, α_3, and β are represented. The α services are performed by a single server, and the β service is represented by a second single server. It is assumed that no two α services can be performed con-

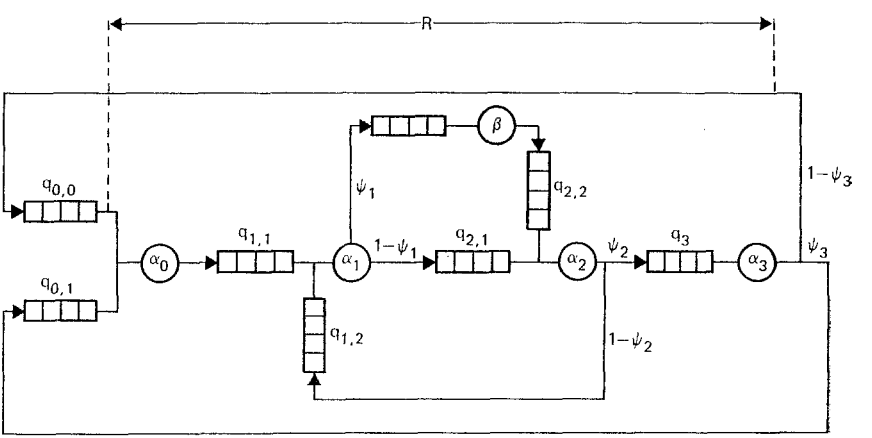

Fig. 4. IMS DL/I component model.

(i) Processor renders α_0, α_1, α_2, and α_3 services.
(ii) I/O unit renders β service.
(iii) α_0, α_1, α_2, and β services are not interruptable.
(iv) α_3 service has preemptive resume type interruption at completion of β service.
(v) Processor scheduled according to priority ordering of queues $q_{0,0}$, $q_{0,1}$, $q_{1,1}$, $q_{1,2}$, $q_{2,1}$, $q_{2,2}$, and q_3.
(vi) Routing determined by binary valued random variables ψ_1, ψ_2, and ψ_3.

currently, but that any α service can be performed concurrently with a β service. Each of α_0, α_1, α_2, and β services is assumed to be noninterruptable. The α_3 service, however, is interruptable at the completion of a β service, this interruption being of the preemptive-resume type. A fixed number of prioritized jobs, denoted 1, 2,..., J (in order of decreasing priority) circulate in the network from time zero. In each of the α queueues service is rendered according to job priority. Jobs are routed through the network in accordance with the binary-valued variables ψ_1, ψ_2, and ψ_3. The epoch of completion of any α service, or the epoch of completion of a β service at which either no α service is in progress or an α_3 service is in progress is called a scheduling decision epoch. The next α service to be initiated, if any, is determined by a scheduling algorithm that employs a total ordering of the queues: $q_{1,2}$, $q_{2,1}$, $q_{1,1}$, $q_{2,2}$, $q_{0,1}$, $q_{0,0}$, q_3.

This queueing network is a representation at the transaction level of the DL/I component of an Information Management System (IMS) database management system (see ref. 10). It is based on the model given by Lavenberg and Shedler,[13] to which the reader is referred for a detailed discussion of the system context. Briefly, the α services are rendered by a processor and the β services by an I/O unit. A transaction against the database, executing in one of J application regions, given rise to a sequence of DL/I calls, each of which results in an access path search of physical blocks until the target segment is found. In the network a job is a transaction executing in a particular region, and the routing labeled $1 - \psi_2$ is interpreted as a further search for the target segment of a DL/I call; similarly, the routing labeled ψ_3 corresponds to an additional DL/I call for a transaction. The routing ψ_1 is interpreted as a block exception, i.e., an I/O operation to the data base to retrieve a required path block. Of interest in this model are the response times for transactions.

Associate with the queueing network of Fig. 4 a marked multivariate point process related to transaction response times as follows. Define the start of a transaction to be the epoch at which processing to determine the first path segment to be accessed for the first DL/I call begins; these epochs correspond in the queueing network to the start of the first α_0 service for the transaction (cf., ref. 13, p. 441). View an event of type a_j as occurring when job j begins α_0 service from $q_{0,0}$ and view and event of type b_j as occurring when, upon completion of α_3 service, job j enters $q_{0,0}$. Identify $W_n^{(j)}$ as the number of times the nth job j enters $q_{0,1}$ between its $(n-1)$st and nth entrance into $q_{0,0}$. Then $R_n^{(j)}$ corresponds to the response time of the nth transaction executed in region j. Note that the workload mark $W_n^{(j)}$ corresponds to the number of DL/I calls in the nth transaction executed in region j. Other definitions of workload measures requiring vector-valued marks will be discussed below.

Common to all three examples is that there are several concurrent job streams contending for service, and that this contention directly affects the response time of a job. Job response time in the network representation is a random sum of queueing times, where components of the random sum are determined by the routing of the job through the network. Aspects of the job routing, including the number of times a particular path is taken, can be thought of as characteristics of the load placed on the system by the job. The *R*-process model defined in Sec. 3 is an abstraction, focusing on response times, of the contention among job streams in which starts and ends of individual job response times are events, and the event marks carry the workload information related to job routing. In the next section we give an interval specification of the *R*-process model.

3. SPECIFICATION OF THE *R*-PROCESS MODEL

The counting specification of a multivariate point process is in principal fundamental; thus, for example, a bivariate point process of type a events and type b events is specified by $\mathcal{N} = \{N(t_1, t_2): t_1, t_2 \geqslant 0\}$ where

$$N(t_1, t_2) = (N_a(t_1), N_b(t_2))$$
$$N_a(t_1) = \text{cumulative number of type } a \text{ events in } (0, t_1]$$

and

$$N_b(t_2) = \text{cumulative number of type } b \text{ events in } (0, t_2]$$

An interval specification of the process is often more convenient; see Cox and Lewis[5] for a detailed discussion of the specification of dependence and correlation in multivariate point processes.

Definition 1. Let J be a positive integer and for $1 \leqslant j \leqslant J$, let sequences of real, possibly vector-valued constants $\{w_n^{(j)} : n \geqslant 1\}$ be given. Then an MMPP having (mutually exclusive) event types $a_1, a_2, \ldots, a_J, b_1, b_2, \ldots, b_J$ is an *R-process* provided that conditions 1–6 hold.

1. In the marginal bivariate point process $\mathcal{N}^{(j)} = \{N^{(j)}(t_1, t_2): t_1, t_2 \geqslant 0\}$ of type a_j events and type b_j events, the event types alternate. For $1 \leqslant j \leqslant J$ and $n \geqslant 1$, denote by $S_n^{(j)}$ the time of the nth type a_j event in $\mathcal{N}^{(j)}$ following the time origin. Also, denote by $T_n^{(j)}$ the time of the nth type b_j event after time $S_1^{(j)}$. Define $T_0^{(j)}$ to be the time of the first type b_j event following the time origin provided that it occurs before time $S_1^{(j)}$, and 0 otherwise.

2. For $1 \leqslant j \leqslant J$ and $n \geqslant 1$, $w_n^{(j)}$ is the mark on the nth event in $\mathcal{N}^{(j)}$ after the time origin; if this nth event is of type a_j, then $w_{n+1}^{(j)} = w_n^{(j)}$.

3. For $1 \leqslant j \leqslant J$ and $n \geqslant 1$,

$$P\{T_n^{(j)} - S_n^{(j)} \leqslant t \mid [T_m^{(k)}, S_l^{(k)}, W_l^{(k)}; 0 \leqslant m \leqslant N_b^{(k)}(S_n^{(j)}), \\ 1 \leqslant l \leqslant N_a^{(k)}(S_n^{(j)})]; 1 \leqslant k \leqslant J\} \\ = P\{T_n^{(j)} - S_n^{(j)} \leqslant t \mid W_n^{(j)}, Z(S_n^{(j)})\}$$

where for $t \geqslant 0$,

$$Z(t) = f(S^{(1)}_{N_a^{(1)}(t)}, T^{(1)}_{N_b^{(1)}(t)}, W^{(1)}_{N_a^{(1)}(t)}, \ldots, S^{(J)}_{N_a^{(J)}(t)}, T^{(J)}_{N_b^{(J)}(t)}, W^{(J)}_{N_a^{(J)}(t)})$$

for f a real, possibly vector-valued function.

4. For $1 \leqslant j \leqslant J$, independent of n,

$$P\{T_n^{(j)} - S_n^{(j)} \leqslant t \mid W_n^{(j)} = w, Z(S_n^{(j)}) = z\} = F^{(j)}(t; w, z)$$

5. For $1 \leqslant j \leqslant J$ and $n \geqslant 1$,

$$P\{S_{n+1}^{(j)} - T_n^{(j)} \leqslant t \mid [T_m^{(k)}, S_l^{(k)}, W_l^{(k)}; 0 \leqslant m \leqslant N_b^{(k)}(T_n^{(j)}), \\ 1 \leqslant l \leqslant N_a^{(k)}(T_n^{(j)})]; 1 \leqslant k \leqslant J\} \\ = P\{S_{n+1}^{(j)} - T_n^{(j)} \leqslant t \mid Z(T_n^{(j)})\}$$

where $Z(\cdot)$ is defined in condition 3.

6. For $1 \leqslant j \leqslant J$, independent of n,

$$P\{S_{n+1}^{(j)} - T_n^{(j)} \leqslant t \mid Z(T_n^{(j)}) = z\} = G^{(j)}(t; z)$$

By condition 1, an R-process can be thought of as being built up by superposition of J (dependent) marginal bivariate point processes having alternating event types. Condition 2 says that in each of these J marginal processes the event marks on a type a event and the immediately following type b event are identical. Dependence of the J marginal processes is specified by conditions 3–6. Condition 3 says that in the jth marginal process the time from a type a_j event to the next (type b_j) event depends on the mark on the type a_j event and also on the past history of the process; dependence on the past is only through the value of Z at the time of the type a_j event. The function f in the definition of Z restricts the dependence on the past to the times of the last type b events and the times and marks on the last type a events in the other processes. As discussed in Sec. 4, the f function permits the representation of the effects of contention among different job streams. Condition 5 is similar to condition 3 in that it restricts the dependence on the

past of the time from a type b_j event until the next (type a_j) event. Conditions 4 and 6 are time homogeneity conditions. Observe that in accordance with Definition 1 the structure of an R-process is specified by J, the function f, and probability distribution functions $F^{(1)},\ldots, F^{(J)}$ and $G^{(1)},\ldots, G^{(J)}$. For generation of an R-process beyond a fixed time origin, specification of initial conditions is also required.

Note that in the definition of the R-process, we have made no stochastic assumptions concerning the sequences of event marks $\{w_n^{(j)}: n \geqslant 1\}$, and have defined the process conditionally on the event marks. In particular, no assumption of stationarity of event mark sequences has been made. The conditional definition of R-process that has been given permits us (as in Sec. 5) to consider response times resulting from deterministic sequences of event marks derived from system traces.

An R-process is well defined by Definition 1 in the sense that, given sequences of event marks $\{w_n^{(j)}: n \geqslant 1\}$, the total series of marked, typed events is determined. To show this we shall outline a method for generating the R-process; this provides a basis for estimation of job response times by simulation of the R-process model. The procedure given in Sec. 3.2 generates sequences of times-to-events $\{S_n^{(j)}\}$ and $\{T_n^{(j)}\}$ on a fixed interval $(0, t_0]$, given sequences of event marks $\{w_n^{(j)}\}$. The total series of events comprising the R-process is obtained by superposition of the events occurring at $\{S_n^{(j)}\}$ and $\{T_n^{(j)}\}$. It is assumed in Sec. 3.2 that the first event after the time origin in the jth marginal process is of type a_j and that it occurs at given times $S_1^{(j)}$. It is also assumed that the times and marks of the last two events in each of the j marginal processes prior to the time origin are given.

The following notation is required:

1. For $1 \leqslant j \leqslant J$, $I^{(j)} = 1$ if the last event prior to time t in the jth marginal bivariate point process is of type a_j, and 0 otherwise.

2. $N_a^{(j)}$ and $N_b^{(j)}$ are counters of the numbers of events of types a and b generated for marginal process j and $N^{(j)} = N_a^{(j)} + N_b^{(j)}$.

3. For $1 \leqslant j \leqslant J$, $Y^{(j)}$ is the time of the last event generated in the jth marginal process.

3.1. Generation Method

1. Initialize for $1 \leqslant j \leqslant J$: $I^{(j)} = 1$, $Y^{(j)} = S_1^{(j)}$, $N_a^{(j)} = 1$, $N_b^{(j)} = 0$, $N^{(j)} = 1$.
2. Let k be the index of $\min\{Y^{(1)},\ldots, Y^{(J)}\}$. If $I^{(k)} = 0$, go to 4;
3. Set $I^{(k)} = 0$, $N_b^{(k)} = N_b^{(k)} + 1$. Generate X having distribution $F^{(k)}(t; w_{N_a^{(k)}}^{(k)}, Z(Y^{(k)}))$. Set $T_{N_b^{(k)}}^{(k)} = Y^{(k)} + X$. Go to 5;

4. Set $I^{(k)} = 1$, $N_a^{(k)} = N_a^{(k)} + 1$. Generate X having distribution $G^{(k)}(t; Z(Y^{(k)}))$. Set $S_{N_a^{(k)}}^{(k)} = Y^{(k)} + X$;

5. Set $N^{(k)} = N_a^{(k)} + N_b^{(k)}$, $Y^{(k)} = Y^{(k)} + X$. If $t_0 \geqslant \min\{Y^{(1)}, \ldots, Y^{(J)}\}$, go to 2. Otherwise, exit.

4. SELECTION OF PARAMETERS IN AN *R*-PROCESS

In accordance with Definition 1 given in the preceding section, an *R*-process is specified by

1. a positive integer J, determining the number of marginal bivariate point processes;

2. a real, possibly vector-valued function f determining the nature of the dependence on the past of conditional times between events in the marginal processes;

3. for $1 \leqslant j \leqslant J$, a distribution function $F^{(j)}(t; w, z)$ for the conditional time from an event of type a_j to the next type b_j event; and

4. for $1 \leqslant j \leqslant J$, a distribution function $G^{(j)}(t; z)$ for the conditional time from an event of type b_j to the next type a_j event.

In this section we indicate ways in which these inputs to the *R*-process model can be selected, given a queueing network system representation.

The selection of a value for J in the *R*-process is straightforward, e.g., for the queueing networks of Examples 1, 2, and 3, J is, respectively, the fixed number of jobs, finite capacity of the network, and maximum number of active regions. Selection of a function f to represent the effects of contention among job streams, however, is more involved. We give several examples.

Example 4. Consider the queueing network of Example 1 in which an event of type a_j (resp. b_j) is the start (resp. termination) of a response time for job j. A response time for job j depends not only on the number of center 1 services it receives (the mark $W_n^{(j)}$) and their durations, but also on the durations of services for other jobs for which it waits in queue. In particular, the response time depends on the number of jobs ahead of job j each time during the response time that job j enters the queue in center 1. As a simple representation of the effects of this complex queueing phenomenon, we might define the f function in the *R*-process by

$$f(s_1, t_1, w_1, \ldots, s_J, t_J, w_J) = \sum_{k=1}^{J} I(s_k - t_k)$$

where $I(x)$ equals 1 if $x \geqslant 0$ and 0 otherwise. Then $Z(S_n^{(j)})$ is precisely the number of jobs at center 1 when the nth response time for job j starts.

Example 5. Consider the queueing network of Example 3 in which an event of type a_j (resp. b_j) is the start (resp. termination) of a transaction in application region j and $W_n^{(j)}$ is the number of DL/I calls in the nth transaction executed in region j. The response time for a transaction in region j depends in a complex manner on the number of other regions executing transactions, the number of DL/I calls and segment searches in these transactions, the pattern of block exceptions and scheduling priorities, etc. As a simple representation of the effects of this resource contention we might define

$$f(s_1, t_1, w_1, \ldots, s_J, t_J, w_J) = \sum_{k=1}^{J} w_k I(s_k - t_k)$$

Then $Z(S_n^{(j)})$ is the total number of DL/I calls in transactions active at the start of the nth transaction in region j.

Alternatively, take $W_n^{(j)}$ to be the total number of access path segment instances searched for the nth transaction in region j. To take into account the effect of priority scheduling of regions, we might define a vector-valued f function

$$f(s_1, t_1, w_1, \ldots, s_J, t_J, w_J) = \mathbf{z} = (z_1, \ldots, z_J)$$

where

$$z_j = \sum_{k=1}^{j-1} w_k I(s_k - t_k)$$

Here z_j is the total number of access path segments searched in transactions active at the start of the nth transaction in region j and executing in a region having higher priority. The distribution function $F^{(j)}(t; w, z)$ would depend on $\mathbf{z}$ only through z_j.

Selection of the distribution function $F^{(j)}(t; w, z)$ and $G^{(j)}(t; z)$ for conditional times between events can be approached in the following way. Recall, for example, that for $1 \leqslant j \leqslant J$,

$$F^{(j)}(t; w, z) = P\{T_n^{(j)} - S_n^{(j)} \leqslant t \mid W_n^{(j)} = w, Z(S_n^{(j)}) = z\}$$

and denote the mean function of this distribution by $\mu_F^{(j)}(w, z)$. This quantity $\mu_F^{(j)}(w, z)$ is the mean time from an event of type a_j until the next type b_j event, given that the mark on the type a_j event is w and z is a measure of the dependence of this time between events on the past. The distribution

$F^{(j)}(t; w, z)$ is probably then most easily specified by restriction to a single standard form for the distribution (e.g., exponential, gamma, mixed exponential) and then providing mean functions and perhaps other necessary functions (e.g., for variance) to reflect the conditioning variables.

Given a queueing network system representation showing the flow of jobs through the network, mean values of the time to traverse a portion of the network can generally be obtained from (presumed) known information (e.g., service time, routing) when there is only a single job in the network. The effect of contention among jobs is to increase this mean time. The conditional mean time between events $\mu_F^{(j)}(w, z)$ that we must specify in the R-process can be chosen by carrying out a single-job flow calculation with respect to the queueing network representation, and then making an additional assumption about the nature of the time increase. Assume that the measure z increases with contention, and let z_0 be the value of z corresponding to no contention. One way to proceed is to assume a product form for $\mu_F^{(j)}(w, z)$, i.e.,

$$\mu_F^{(j)}(w, z) = g_F^{(j)}(z)\, h_F(w)$$

where $h_F(w) \geqslant 0$, $g_F^{(j)}(z_0) = 1$ and $dg_F^{(j)}/dz \geqslant 0$. The last two constraints ensure consistency and guarantee that the effect of contention is to increase the mean time between events. Other models, e.g., an additive form, can also be considered.

Note that an explicit assumption [e.g., the function $g_F^{(j)}(z)$] about the effect of contention among job streams is an input to an R-process model. This is to be contrasted with a queueing network model in which the assumption is made implicitly when particular queueing-theoretic (e.g., independent, identically distributed) stochastic assumptions are put forth.

Example 6. Consider the queueing network of Examples 1 and 4. Here $W_n^{(j)}$ corresponds to the number of center 1 services received by job j during its nth response time. Then, for $1 \leqslant j \leqslant J$, $h_F(w) = E\{\alpha\}\, w$, where $E\{\alpha\}$ is the mean center 1 service time. If, as in Example 4, $Z(S_n^{(j)})$ corresponds to the number of jobs enqueued at center 1 when the nth response time for job j begins, then $z_0 = 1$ and a plausible assumption in the R-process model might be that $g_F^{(j)}(z) = 1 + c(z - 1)$ for some nonnegative constant c.

5. ANALYSIS OF RESPONSE TIME DATA

In this section we report the analysis of transaction processing data obtained from a running database system and observe its conformity to an R-process model. The database system considered is the online portion of

IMS/VS 1.0.1[10] running under OS/VS Release 1.6 on an IBM System 370/145 with 1 megabyte main storage. The particular configuration of IMS studied here has two online message processing regions and a 100K byte buffer pool. The database is from an IMS manufacturing production control system running on an IBM 370/155, and the workload is a re-creation of a day's transaction stream from that system, based on a recorded trace. In re-creating the workload of the original system, activity outside the DL/I component (database access section) of IMS is minimized in the following way. In the original system instances of a number of transaction types occur, each of which is processed by a corresponding application program that must be scheduled and loaded in a message processing region. In the experimental system this workload is re-created by initially loading each message processing region with a program that processes messages from a message queue pre-loaded for that region. The messages in the queues consist of a sequence of transaction header messages, each followed by messages containing the exact sequence of DL/I calls issued by a particular transition instance as it actually occurred in the original system. The program in each region merely receives these messages and issues the DL/I calls. In this sense the experimental system runs in a fully saturated mode, and the measured response times consist essentially of time spent in the DL/I component of IMS.

The processing by IMS of the entire day's re-created transaction stream was traced, and the times of transaction instance start and end were recorded along with message processing region, transaction type, and counts of DL/I calls and access path segments searched.

After examination of the gross features of the data for the entire day, a serial section of length $t_0 = 1800$ (in unspecified time units) was selected for detailed analysis. This section showed a relatively high transaction completion rate in both regions with a correspondingly low average response time, as well as relatively low average numbers of DL/I calls and access path segments searched per transaction instance. The section contained 1490 transaction instances in region 1 and 394 in region 2.

In examining the data to get some indication of the overall behavior of the system during the selected time period, an immediate observation was the inhomogeneity of the several univariate point processes of transaction completions. Figure 5 is a plot of the cumulative number of transaction completions, along with the cumulative number for each of the regions individually, versus time. In a stationary point process the cumulative-number-of-events plot would be close to a straight line. Here there is an indication of some fluctuation in the transaction completion rate throughout the section and a definite decrease toward the end. This inhomogeneity could be caused by a blocking phenomenon within the DL/I component or

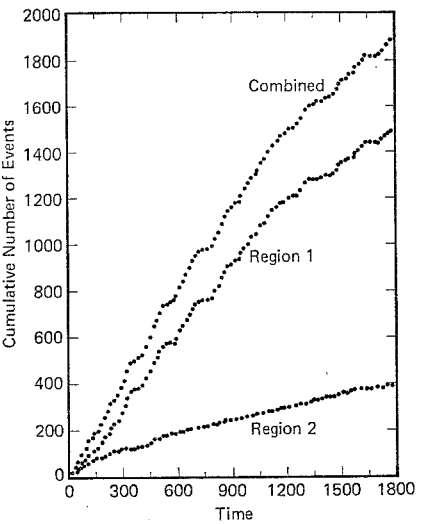

Fig. 5. Cumulative numbers of transaction completions.

be a direct effect of a changing transaction mix. A similar inhomogeneity was observed in the original system (see ref. 15).

Next we examined the transaction response times themselves. Table I gives sample statistics for sequences of response times in each region, sectioned serially into 10 sections in region 1, and five sections in region 2. It can be seen that the characteristics of the sections differ widely, indicating that the response times are not homogeneous in time. Similar sectioning and calculation of sample statistics was performed for the transaction workload measures: the number of DL/I calls and access path segments searched per transaction instance. Comparison of these statistics with those for response times in each region showed that to a great extent the response times mimicked changes in the workload measures. This is illustrated in Fig. 6, a plot of the means of the response times and the workload measures of the transactions in region 1 for each of the 10 sections on a common scale that gives their overall mean the same order of magnitude. This type of behavior was also observed in the other sample statistics such as variance. This suggests in a general way that a model of the R-process type, where response times reflect variations in workload characteristics, may be appropriate.

Accordingly, the focus of the remaining data analysis is on providing the necessary concomitants of an R-process model. We consider R-process models in which the bivariate point processes are the sequence of transaction starts and ends in the $J = 2$ message processing regions. The workload marks on the events are taken to be vector-valued, consisting of the

Table I. Sample Characteristics of Response Times in Serial Sections

Section	Sample mean ($\bar{X}$)	Sample variance (S^2)	S.D. of mean ($S_{\bar{X}}$)	Coefficient of variation ($S/\bar{X}$)	Coefficient of skewness ($\hat{\beta}_1$)	Coefficient of kurtosis ($\hat{\beta}_2$)
			Region 1: Section size = 149			
1	0.9784	0.9544	0.0800	0.9985	2.1934	9.2259
2	0.6233	0.6022	0.0636	1.2450	4.2741	24.9627
3	0.7565	6.3256	0.2060	3.3246	8.6298	84.2965
4	0.8175	4.1492	0.1669	2.4917	9.3442	101.6443
5	0.5316	0.3421	0.0479	1.1003	6.6353	60.8537
6	0.9769	5.9130	0.1992	2.4893	5.9178	38.7558
7	0.8246	1.0440	0.0837	1.2391	4.7371	32.9017
8	0.9266	2.7614	0.1361	1.7934	6.2700	47.1030
9	1.7174	3.7906	0.1595	1.1337	1.2191	3.1611
10	1.6866	35.3707	0.4872	3.5261	11.0734	130.1037
			Region 2: Section size = 78			
1	1.6867	1.6400	0.1450	0.7592	3.6967	23.0232
2	3.2475	53.1111	0.8252	2.2441	6.1901	44.2911
3	4.1003	13.6466	0.4183	0.9009	2.4918	11.9608
4	5.0246	19.1029	0.4949	0.8699	2.0771	8.7578
5	5.2204	23.5742	0.5498	0.9301	3.4759	19.9693

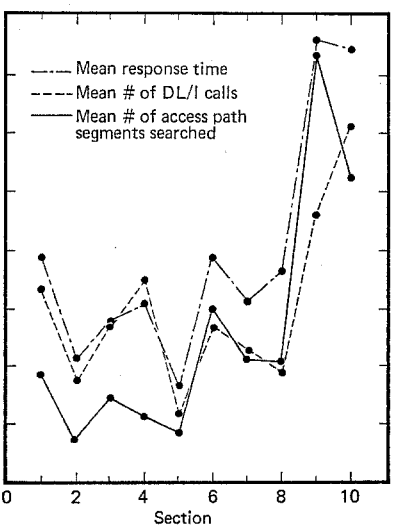

Fig. 6. Means of transaction response times and workload measures in region 1 by section.

counts of DL/I calls and access path segments searched for each transaction instance. Based on this structure we consider distributional forms for the conditional response times and the conditional intertransaction times, and the specification of their dependence on the workload measures and the past history of the process.

We examine the transaction response times first. It is clear conceptually that a response time is a sum of the times to initiate each contained DL/I call (e.g., interpret the search argument) and to search the segments in each call's access path. Therefore, since the available workload mark on each transaction ($W = (X_1, X_2)$, say) gives the number of DL/I calls (X_1) and access path segments searched (X_2) in the transaction, the response time conditional on the workload mark might be expected to have a mean value that is approximately linear in the components of the mark.

It is also clear that the past history of the transaction processing stream could affect the response time. In particular, a transaction that runs in one of the message processing regions while the other region is inactive can be expected to fill the buffer with its required blocks, to the detriment of the response time of a transaction starting subsequently in the other region.

In the R-process model such contention may be made explicit through the process $\{Z(t)\}$ defined in Sec. 3.1 which is a possibly vector-valued function f of the latest times of transaction starts and terminations and the latest workload marks in all regions at time t. For the two-region example considered here, the f function may be written as $f(s_1, t_1, w_1, s_2, t_2, w_2)$. To indicate the type of contention discussed, we define f to be vector valued with components (f_1, f_2) defined by

$$f_1(s_1, t_1, w_1, s_2, t_2, w_2) = I(s_2 - t_2) \min(s_1 - s_2, s_1 - t_1) + I(t_2 - s_2)\, I(t_2 - t_1) \min(t_2 - s_2, t_2 - t_1)$$

and f_2 the same with the one's and two's interchanged on the right-hand side. The jth component of the function f is, at the start of a transaction in region j, the amount of time since the most recent transaction termination in region j that the most recent transaction in the other region has run. During this time the transaction in the other region is executing by itself; therefore, the greater this time, the greater its expected effect on the response time of the transaction about to start in region j. A simple way of incorporating this dependence explicitly is to let an expected response time in region j, conditioned on $Z(t)$ at the transaction start, increase linearly with the jth component of Z.

Accordingly, a model for the mean response time $R_n^{(j)}$ of the nth trans-

action in the jth region, conditional on the workload mark $W_n^{(j)}$ and on $Z(S_n^{(j)})$, is

$$E\{R_n^{(j)} \mid W_n^{(j)} = (x_1, x_2), Z(S_n^{(j)}) = (x_3^{(1)}, x_3^{(2)})\}$$
$$= \beta_0^{(j)} + \beta_1^{(j)} x_1 + \beta_2^{(j)} x_2 + \beta_3^{(j)} x_3^{(j)} \tag{5.1}$$

where the β's are unknown parameters. The parameter β_0, in effect, gives the average overhead time for transaction processing; β_1 and β_2 give the average increase in response time due to an additional DL/I call or access path segment searched; and β_3 gives the average penalty in response time for each unit of time that the prior transaction in the other region ran alone.

Assuming that the conditional response times are uncorrelated random variables with means given by (5.1) and equal variances, the unknown parameters in the conditional expectation may be estimated from the data by ordinary least squares methods. This has been done for each region, and ostensibly the model (5.1) for the mean fits well since it explains almost all the variation in the response times as indicated by the squared multiple correlation coefficient. However, plots of residuals from the fitted model reveal that in each region the variation of the residuals increases with the magnitude of the predicted expected value. The increase appears to be such that the residual variance is approximately proportional to the square of the predicted expected response time. Such an inequality of the variance lessens the efficiency of the least squares parameter estimators (which are, however, still unbiased), and wholly distorts the usual estimators of their standard errors. An estimation procedure suitable for this situation is iterative weighted least squares (IWLS) which is described in Appendix 1 (cf., Bradley[2] or Charnes *et al.*[4]).

Estimation by this method was done for each region, and examination of the weighted residuals indicated that the variance had been equalized, a confirmation of the procedure. The estimated parameters for each region are given in Table II, where the fourth column gives the estimated coefficient divided by its estimated standard error. This t statistic may be treated as approximately normally distributed, and its absolute value indicates how significantly different from zero the coefficient is. All the estimated parameters for both regions are significantly different from zero with the exception of the constant for region 2. Though this estimate is actually negative and could legitimately be dropped from the model in region 2 due to its insignificance, we retain it for consistency with the observation that it causes no problems, since every transaction must have at least one DL/I call and search one path segment; thus no negative response times will be predicted. Note also that the estimated parameters for region 1 are generally smaller in magnitude than the corresponding estimates for region 2, apparently a

Table II. IWLS Parameter Estimates for the Conditional Mean Response Times

Coefficient	Estimate	Estimated S.E.	t
Region 1			
Constant	.0505	.0060	8.44
No. DL/I calls	.0696	.0025	28.10
No. access path segments	.0048	.0002	21.88
Contention measure	.1799	.0285	6.31
Constant of proportionality IWLS: .1815 ML: .1650			
Region 2			
Constant	−.0385	.0708	−.54
No. DL/I calls	.3635	.0160	22.65
No. access path segments	.0053	.0014	3.71
Contention measure	.2929	.1324	2.21
Constant of proportionality IWLS: .2399 ML: .2001			

reflection of the different scheduling priorities between the regions. The model for the means and variances has been further checked, particularly using residual plots to verify the linear form of the mean, and has been judged to represent the data adequately.

Finally, a complete specification of the R-process model requires a particular conditional response time distribution function incorporating the observed mean and variance function. Since the response times are positive random variables with variance proportional to the mean squared, a number of standard families of distributions with these properties, such as the lognormal and gamma, are immediate candidates. Examination of the response times revealed that a gamma distribution with mean specified by (5.1) and a constant shape parameter provided an adequate fit in each region. Since the parameterization of a gamma distribution in terms of its mean, and fitting a linear model for this mean may not be well known, we present details in Appendix 2.

Under the gamma assumption the maximum likelihood (ML) estimates of the parameters in (5.1) are the same as the IWLS estimates; therefore they have the values exhibited in Table 2. The ML estimates of the constants of proportionality are also given in Table 2 and, since they are smaller than the IWLS estimates, the estimated standard errors of the coefficients are slightly smaller and the t statistics slightly greater under the gamma assumption. The gamma assumption itself was examined through gamma probability plots of the normed response times (i.e., observed divided by fitted). Such a plot should fall along a straight line of slope 1 through the origin, with larger departures possible in the tails, if the gamma assumption is correct.

It was observed, as for instance, in the plot for region 1, which is given in Fig. 7, that the gamma parametric model with the mean function (5.1) and the estimated parameters of Table 2 appears to describe the conditional response time distribution well.

The intertransaction times in each region were considered in the same fashion as the response times, and it was observed that the workload marks had no discernible influence. A number of plausible measures of the effects of contention on these times were also considered. What appeared to be significant was a vector-valued function with two components whose jth component at the end of a transaction in the jth region is defined as follows. If there was no transaction active in the other region, the value is zero; if there was a transaction active, the value is equal to the time that this transaction ran simultaneously with the prior transaction in the jth region. We believe that this function gives a measure of how long the other transaction will hold the inner loop of the DL/I component, thereby delaying the start of a new transaction in the jth region.

Thus, formally, the $\{Z(t)\}$ process described here for the response times is expanded to a four-component vector containing the above function for each region as its third and fourth components. The conditional expectations of the intertransaction times were modeled as linear in this contention measure, and again gamma models for these times, using the IWLS estimates of the unknown parameters, fit the data adequately. This provides the formal specification of the conditional distribution of the times between events required for the R-process model.

With the selection of the conditional distribution of the response times and intertransaction times in each region, we have completed the specification of an R-process model. We now consider the overall goodness of fit of this model. Formal tests of the dependence structure in an MMPP, and in particular for an R-process, are not available. However, we can address the good-

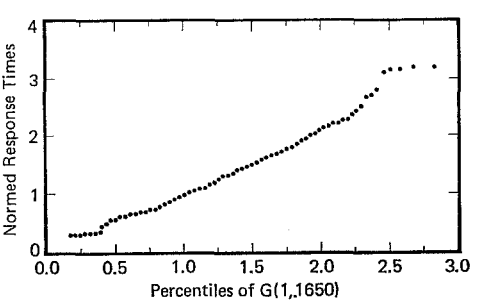

Fig. 7. Gamma probability plot of normed response times from region 1.

ness of fit by informal graphical comparisons of characteristics of the original point process data with the corresponding characteristics of the fitted R-process model.

The procedure used to select forms for an estimate parameters of the conditional distributions of response times, and our examination of the residuals ensure a good fit of the R-process model in terms of the intervals (times between events) in the marginal bivariate point processes. However, the interaction of these specifications to produce an MMPP that resembles the data (including its inhomogeneities) is not implicit in the procedure. One way to examine this aspect of the fit of the model is to compare the expected values over time of the various counting processes in the fitted model with their empirical counterparts from the data. Note that dependence between the marginal bivariate point processes in the R-process is reflected in these counting processes.

We compare graphically, for each of the regions and for both combined, the expected number of transactions completed in the fitted R-process model and the number observed in the original data as a function of time. To do so requires the calculation for the R-process model, using the estimated parameters, of the quantities $E\{N_b^{(1)}(t)\}$ and $E\{N_b^{(2)}(t)\}$ for t in $(0, t_0]$. These calculations appear to be very difficult, and instead we resorted to simulation of the R-process model. Using APL implementations of algorithms for generation of random numbers given by Robinson and Lewis[18] we made 20

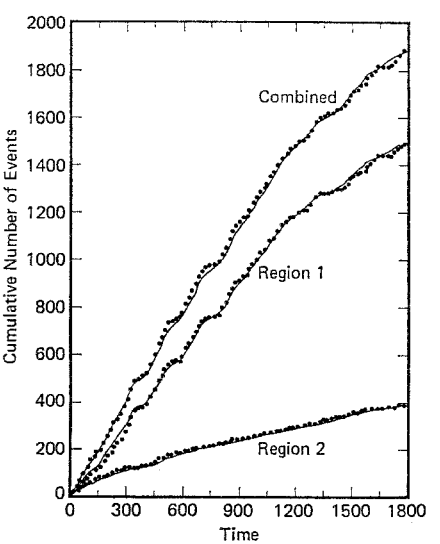

Fig. 8. Cumulative numbers of transaction completions for data and R-process model.

replications of the specified model on the interval (0,1800]. From these replications the values of the several counting processes were collected at 100 equally spaced time points in the interval. At each of these time points, the sample mean of the counts from the 20 replications was computed to provide a point estimate of the expected value there. In Fig. 8 these point estimates of the mean numbers of transaction completions are plotted (as solid lines) for each region and both regions combined, along with numbers (dots) for the original data from Fig. 5. It is clear that, based on these counting processes, the *R*-process model closely approximates the data and reflects its constituent inhomogeneities.

6. CONCLUDING REMARKS

The *R*-process model introduced in this paper provides a framework for the formal description of the response time behavior of multiprogrammed systems. The *R*-process privides a high-level model for prediction of characteristics of job response times for specified workloads. Prospective use of the *R*-process model involves the setting down of a (possibly quite detailed) queueing network representation of the system being modeled; this representation serves as the basis for the formulation of the *R*-process model and for the selection of parameters therein.

The specification of the structure of the *R*-process conditional on the workload marks effectively separates the difficult problem of formal representation of workload characteristics from the overall problem of response time prediction. In particular, note that we need not have a formal model for the workload marks in order to use the *R*-process model; data sequences obtained from system traces can be used directly.

Although the analysis of data reported in this paper is from a single database management system, we feel that the type of gross inhomogeneities observed in this system (primarily caused by a rapidly changing transaction mix) are likely to be found in other systems. Performance studies that fail to reflect the inhomogeneous behavior of running systems can be misleading; the *R*-process model provides a convenient way of approaching this aspect of system performance prediction.

Several directions for further work are apparent. Methods for the efficient simulation of general characteristics of response times in *R*-processes need to be developed. In this connection simulation experience with particular *R*-process models and related queueing network models would be of interest.

Formulation of more general marked multivariate point process models (e.g., allowing multivariate marginal processes along with more general patterns of event types) can be considered. Such extensions of the *R*-process model have application to performance prediction for database management

system in that more detail of the pattern of access to the database can be represented.

APPENDIX 1

Let $\mathbf{Y}$ be an $N \times 1$ dimensional random vector and $\mathbf{X} = (x_{ij})$ be an $N \times K$ matrix of constants and random variables, such that

$$E(\mathbf{Y} \mid \mathbf{X}) = \mathbf{X}\boldsymbol{\beta}$$

for $\boldsymbol{\beta}$ a $K \times 1$ vector of unknown parameters, and

$$\mathrm{Cov}(\mathbf{Y} \mid \mathbf{X}) = \boldsymbol{\Gamma} = \mathrm{Diag}(\sigma_1^2, \ldots, \sigma_N^2)$$

where the σ_i^2 are N unknown positive scalars that give the variances of the components of $\mathbf{Y}$. If the σ_i^2 were known up to a constant of proportionality, the weighted least squares (WLS) estimator of $\boldsymbol{\beta}$ for a given observation $\mathbf{y}$ of $\mathbf{Y}$ could be found by solving the linear system

$$\mathbf{X}'\boldsymbol{\Gamma}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = \mathbf{0}$$

This results in a simple closed form for the WLS estimator which can be calculated through a slight variant of ordinary least squares, which is the special case when all of the σ_i^2 are equal. When the variances are not known but a functional form for each σ_i^2 in terms of the $\{x_{ij} : 1 \leqslant j \leqslant K\}$ and the unknown $\boldsymbol{\beta}$ is known up to a constant of proportionality, an estimate of $\boldsymbol{\beta}$ may be found by solving the system

$$\mathbf{X}'(\boldsymbol{\Gamma}(\boldsymbol{\beta}))^{-1}(\mathbf{y} - \mathbf{Y}\boldsymbol{\beta}) = \mathbf{0}$$

in $\boldsymbol{\beta}$. Here $\boldsymbol{\Gamma}(\boldsymbol{\beta})$ indicates that the variances depend on $\boldsymbol{\beta}$. Since the resulting estimator $\hat{\boldsymbol{\beta}}$ may be calculated through an iterative sequence of weighted lead squares computations, it is termed the iterative weighted least squares estimator. The estimated covariance matrix of $\hat{\boldsymbol{\beta}}$ is $s^2(\mathbf{X}'(\boldsymbol{\Gamma}(\hat{\boldsymbol{\beta}}))^{-1}\mathbf{X})^{-1}$, where s^2 is the IWLS estimator of the constant of proportionality given by

$$\frac{1}{N - K}\, \boldsymbol{\epsilon}'(\boldsymbol{\Gamma}(\hat{\boldsymbol{\beta}}))^{-1}\boldsymbol{\epsilon}$$

and $\boldsymbol{\epsilon} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}$ is the residual vector.

In the data examined in Sec. 5 the variance of the residuals for the conditional response times appeared to increase as the square of the expected value given by (5.1). Thus we have a functional form up to a constant of proportionality and, therefore, IWLS may be used to estimate the unknown parameters.

APPENDIX 2

Let Y be a gamma variate with mean $\mu > 0$ and shape parameter $r > 0$. Then Y has the density function

$$g(y) = \left(\frac{r}{\mu}\right)^r \frac{y^{r-1}}{\Gamma(r)} e^{-ry/\mu} I(y \geqslant 0)$$

and variance μ^2/r. Thus the variance is proportional to the mean squared, and $1/r$ is the constant of proportionality. Denote this distribution as $G(\mu, r)$. Given a number of Y_i independently distributed as $G(\mu_i, r)$, the normalization Y_i/μ_i yields independent variates, identically distributed as $G(1, r)$. If we have N observations $\{y_i\}$ of variates distributed as $G(\mu_i, r)$, where the means μ_i are a linear combination of known constants and unknown parameters [e.g., (5.1)], the IWLS estimator of the unknown parameters is also maximum likelihood (cf., Bradley[2]). However, the IWLS estimator of the variance constant of proportionality is generally different from the more efficient maximum likelihood (ML) estimator $1/\hat{r}$. The estimate $\hat{r}$ may be found by a computational procedure that is equivalent to treating $y_i/\hat{\mu}_i$ as $G(1, r)$ where $\hat{\mu}_i$ is the predicted mean, and performing a usual maximum likelihood calculation. Further results on maximum likelihood estimation yield that the ML estimators of the parameters in the mean are asymptotically normally distributed about the correct values with an estimated covariance matrix as given in Appendix 1 for IWLS estimators, only with the ML estimator of the constant of proportionality replacing the IWLS estimator. Finally, a partial verification of this type of gamma parametric form may be made by plotting the ordered values of $y_i/\hat{\mu}_i$ against the appropriate percentiles of the $G(1, \hat{r})$ distribution (i.e., the ith largest $y_i/\hat{\mu}_i$ plotted versus the $1 - i/(n + 1)$ percentage point) to form a gamma probability plot.

ACKNOWLEDGMENTS

We are indebted to W. G. Tuel, Jr., for providing the response time data analyzed in Sec. 5, and for helpful discussions about the operation of the experimental system.

REFERENCES

1. F. Baskett, K. M. Chandy, R. R. Muntz, and F. G. Palacios, "Open, closed, and mixed networks of queues with different classes of jobs," *J. ACM* **22**:248–260 (1975).
2. E. L. Bradley, "The equivalence of maximum likelihood and weighted least squares estimates in the exponential family," *J. Am. Stat. Assoc.* **68**:199–200 (1973).

3. J. Buzen, "Queueing Network Models of Multiprogramming," Ph.D. thesis, Division of Engineering and Applied Physics, Harvard University, Cambridge, Massachusetts (1971).
4. A. Charnes, E. L. Frome, and P. L. Yu, "The equivalence of generalized least squares and maximum likelihood estimates in the exponential family," *J. Am. Stat. Assoc.* **71**:169–171.
5. D. R. Cox and P. A. W. Lewis, "Multivariate Point Processes," in *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability*, Vol. III, pp. 401–448. University of California Press, Berkeley, California (1972).
6. D. P. Gaver, "Probability models for multiprogrammed computer systems," *J. ACM* **14**:423–439 (1967).
7. E. Gelenbe and R. R. Muntz, "Probabilistic models of computer systems, Part I (Exact results)," *Acta Inform.* 7:35–60 (1976).
8. D. L. Iglehart and G. S. Shedler, "Regenerative Simulation of Response Times in Networks of Queues," IBM Research Report RJ 1740, San Jose, California (1976). To appear in *J. ACM* **25** (1978).
9. D. L. Iglehart and G. S. Shedler, "Simulation of Response Times in Finite Capacity Open Networks of Queues," IBM Research Report RJ 1886, San Jose, California (1976). To appear in *Opns. Res.* **26** (1978).
10. IBM Corporation, "Information Management System/360, Version 2," General Information Manual GH20-0765, Armonk, New York (1973).
11. J. R. Jackson, "Jobshop-like queueing systems," *Manage. Sci.* **10**:131–142 (1963).
12. A. G. Konheim and M. Reiser, "A queueing model with finite waiting room and blocking," *J. ACM* **23**:328–341 (1976).
13. S. S. Lavenberg and G. S. Shedler, "Stochastic modeling of processor scheduling with application to data base management systems," *IBM J. Res. Dev.* **20**:437–448 (1976).
14. P. A. W. Lewis and G. S. Shedler, "A cyclic-queue model of system overhead in multiprogrammed computer systems," *J. ACM* **18**:119–220 (1971).
15. P. A. W. Lewis and G. S. Shedler, "Statistical analysis of non-stationary series of events in a data base system," *IBM J. Res. Dev.* **20**:465–482 (1976).
16. C. G. Moore III, "Network Models for Large-Scale Time-Sharing Systems," Technical Report No. 71-1, Department of Industrial Engineering, University of Michigan Ann Arbor, Michigan (1971).
17. M. Reiser and H. Kobayashi, "Queueing networks with multiple closed chains: theory and computational algorithms," *IBM J. Res. Dev.* **19**:283–294 (1975).
18. D. W. Robinson and P. A. W. Lewis, "Generating Gamma and Cauchy Random Variables: An Extension to the Naval Postgraduate School Random Number Package," Naval Postgraduate School Report NPS72Ro75041, Monterey, California (1975).

Printed in Belgium