F. R. Moore

Computational Model of a Closed Queuing Network with Exponential Servers

Abstract: A simplification and extension to Gordon and Newell's approach to closed queuing networks is derived. The equations, valid for exponential servers, with one server being a multiserver, greatly reduce the computation time required by the former approach. In addition, simplified equations for the details of the queuing network are derived.

Introduction

In this paper we deal with closed queuing networks of exponential servers. When only one of the queuing stations is a multiserver, we derive simple equations that greatly reduce the computation time required from that of the Gordon and Newell approach [1].

This model often arises in analyzing computer systems when shared serially reusable resources are requested by multiple users. In the Appendix, one such system is analyzed.

The Gordon and Newell equations require summations over the total number of states of the queuing network. When there are N requestors and $M + 1$ stations, this requires a summation over $\binom{N+M}{M}$ states. In our approach, we reduce this to a summation over M functions, each of which is a sum of $N + 1$ factors divided by a product of $M - 1$ factors, thus greatly reducing the computation time.

Assumptions

A queuing network is a set of multiserver queuing stations, as shown in Fig. 1. Each station has a single first-in first-out (FIFO) queue in which arrivals are noted. The ith station has R_i parallel, identical servers, and the first element in the queue always moves to any idle server as long as a server is not busy. Each server in a given station will service a request in a time that has an identical exponential distribution. The ith station has average service time T_{si}.

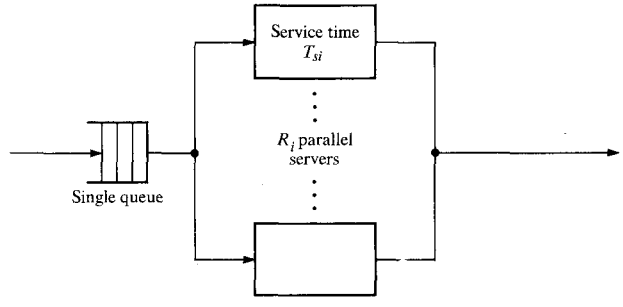


Figure 1 Multiserver queuing station.

Thus the parameters for station i are the service time T_{si} and the number of servers R_i. The stations may be interconnected by an arbitrary, constant, interconnection matrix:

$$P = \begin{Bmatrix} P_{11} & \cdots & P_{1m} \\ \vdots & & \vdots \\ P_{m1} & \cdots & P_{mm} \end{Bmatrix}$$

provided $\sum_{j=1}^{m} P_{ij} = 1$.

Each P_{ij} is the probability that a requestor leaving station i will go to the entrance to station j. Since we specify that the matrix is a constant, this matrix is independent of the state of the system.

A closed queuing network is one in which the total number of requestors, N, circulating in the system is finite and constant. Thus there are no entries or exits from the system and each requestor on completion of his service time proceeds directly, subject to the probability distribution P, to the entry of another queuing station with no transit delay.

Note that if the number of servers R_i of any station is greater than or equal to N, there is never a waiting line and we have just an exponential delay function. On the other hand if any R_i is one, we have a simple single server queue.

Equation derivations

The derivation assumes that there are $M+1$ server stations, with exactly one station, station $M+1$, being a multiserver. All the other M stations are assumed to be simple single servers. The derivation assumes three equations (Eq. (A), (B), and (C) in Ref. 1), which are valid for all multiserver stations. The interested reader is referred to Ref. 1 for notations not explained here and for derivation of these equations.

The derivation begins similarly to one alluded to by Saaty [2], although later portions of the derivation are different.

• *Gordon & Newell's derivation*

The state of the system of server stations is uniquely determined by the $M+1$–tuple

$$\langle n_1, n_2, \cdots, n_M, n_{M+1}\rangle \text{ in which } \sum_{i=1}^{M+1} n_i = N .$$

Each n_i is the number of requestors at the ith station.

Note that the total number of states of the system is the partition of N requestors over $M+1$ stations $\binom{N+M}{M}$.

Now, if we take $p(n_1, \cdots, n_M, n_{M+1})$ as the steady-state probability that the system is in state $\langle n_1, \cdots, n_M, n_{M+1}\rangle$, we can write the standard queuing equations [1]. We define for the kth station the function $\beta_k(n)$ as follows:

$$\beta_k(0) = 1$$

$$\beta_k(n) = \alpha_k(n)\beta_k(n-1)$$

$$\text{where } \alpha_k(n) = \begin{cases} n \text{ if } n \le R_k \\ R_k \text{ if } n \ge R_k . \end{cases}$$

Then the standard queuing equations can be transformed into the following set of equations, as shown in Ref. 1:

$$\sum_{i=1}^{M+1} P_{ik}\,(\mu_i X_i) = \mu_k X_k \qquad k = 1, 2, \cdots, M+1$$

$$\text{or} \quad \underline{y}\,\underline{P} = \underline{y} \tag{A}$$

$$\text{where } \underline{y} = \langle \mu_1 X_1, \cdots, \mu_{M+1} X_{M+1}\rangle$$

and $\mu_i = \dfrac{1}{T_{si}}$.

The X_i are a set of unknown parameters whose solution in the above equation leads us to a solution for $p(n_1, \cdots, n_{M+1})$ as follows:

$$p(n_1, \cdots, n_{M+1}) = \left\{\prod_{i=1}^{M+1} \frac{X_i^{n_i}}{\beta_i(n_i)}\right\} G^{-1}(N) . \tag{B}$$

In this equation $G(N)$ is a normalizing factor that is derived from the fact that all probabilities must sum to one:

$$G(N) = \sum_{\langle n_1, \cdots, n_{M+1}\rangle} \left\{\prod_{i=1}^{M} \frac{X_i^{n_i}}{\beta_i(n_i)}\right\} \tag{C}$$

$$\sum_{i=1}^{M+1} n_i = N .$$

Note that at this point the equations, while solvable, require a large amount of calculation, since Eq. (C) sums over $\binom{N+M}{M} = (N+M)!/M!N!$ different points.

At this point some simplifications are in order to bring computation within bounds.

Further simplifications

We would like to simplify the summation in Eq. (C). To do this we will use a lemma from combinatorial theory and an assumption. We use the assumption that all queuing stations except one are single-server queues. We have $M+1$ servers, and we let $R_i = 1$ for $i = 1, \cdots, M$; $R_{M+1} = R$. Then using the definition of β_i, we find:

$$\begin{aligned} &\beta_i(n_i) = 1 \qquad i = 1, \cdots, M \\ &\beta_{M+1}(n_{M+1}) = \beta(n_{M+1}) \end{aligned} \tag{1}$$

$$\text{or letting } n = \sum_{i=1}^{M} n_i$$

$$\text{then} \qquad n_{M+1} = N - n \tag{2}$$

since we have N requestors.

Now since the solution to (A) is arbitrary to within a constant, we can choose

$$X_{M+1} = 1 .$$

Then Eq. (C) becomes

$$G(N) = \sum_{\langle n_1, \cdots, n_M\rangle} \beta^{-1}(N-n) \prod_{i=1}^{M} X_i^{n_i} \tag{3}$$

$$\sum_{i=1}^{M} n_i \le N .$$

Now we would like to prove a combinatorial theory lemma having to do with the generating function for combinations with repetitions for k distinct objects [3].

Let T_b^k be an enumeration of the ways of distributing k distinct objects into b slots. For example:

$$T_2^3 = X_1^2 + X_2^2 + X_3^2 + 2X_1X_3 + 2X_2X_3$$

is meant to represent that there is one way that each of the three distinct objects, X_1, X_2, X_3 may each be fitted into the two slots, allowing repetitions, and two ways that we may choose one each of the objects to fit into the slots.

Then if t is a dummy enumeration variable, then the generating function for combinations with repetitions for k distinct objects is:

$$GF = \sum_{b=0}^{\infty} T_b^k t^b .$$

Lemma: In the generating function for combinations with repetitions for k distinct objects,

$$T_b^k = \sum_{i=1}^{k} \frac{X_i^{b+k-1}}{\prod_{\substack{j=1 \\ j\neq i}}^{k} (X_i - X_j)} .$$

Proof: By the definition of a generating function we have:

$$GF(k) = \sum_{b=0}^{\infty} T_b^k t^b = \prod_{i=1}^{k} (1 + X_i t + X_i^2 t^2 + \cdots)$$

$$= \prod_{i=1}^{k} \frac{1}{(1 - X_i t)} . \tag{4}$$

Expanding (4) in partial fractions we get:

$$GF(k) = \sum_{i=1}^{k} \frac{X_i^{k-1}}{\prod_{\substack{j=1 \\ j\neq i}}^{k} (X_i - X_j)} \frac{1}{(1 - X_i t)} ,$$

or switching to an infinite summation:

$$GF(k) = \sum_{i=1}^{k} \frac{X_i^{k-1}}{\prod_{\substack{j=1 \\ j\neq i}}^{k} (X_i - X_j)} (1 + X_i t + X_i^2 t^2 + \cdots) .$$

Now collecting the coefficients of t^b we find:

$$T_b^k = \sum_{i=1}^{k} \frac{X_i^{b+k-1}}{\prod_{\substack{j=1 \\ j\neq i}}^{k} (X_i - X_j)} ; \text{ Q.E.D.} \tag{5}$$

Now, we apply this lemma to Eq. (3), noting that we are trying to assign the M single-server stations to the n requestors not at the multiserver station, allowing repetitions. We find:

$$G(N) = \sum_{n=0}^{N} \frac{1}{\beta(N-n)} \sum_{i=1}^{M} \frac{X_i^{n+M-1}}{\prod_{\substack{j=1 \\ j\neq i}}^{M} (X_i - X_j)} . \tag{6}$$

Switching sums:

$$G(N) = \sum_{i=1}^{M} \frac{X_i^{M-1} F(X_i, N, R)}{\prod_{\substack{j=1 \\ j\neq i}}^{M} (X_i - X_j)} , \tag{7}$$

where:

$$F(X_i, N, R) = \sum_{n=0}^{N} \frac{X_i^n}{\beta(N-n)}$$

$$= \frac{1}{R!} \sum_{n=0}^{N-R} \frac{X_i^n}{R^{N-n+R}} + \sum_{n=N-R+1}^{N} \frac{X_i^n}{(N-n)!}$$

$$= \sum_{n=0}^{R} \frac{X_i^{N-n}}{n!} + \frac{1}{R!} \sum_{n=R+1}^{N} \frac{X_i^{N-n}}{R^{n-R}} . \tag{8}$$

Now we can also define

$$H(\underline{X}, i, M) = \frac{X_i^{M-1}}{\prod_{\substack{j=1 \\ j\neq i}}^{M} (X_i - X_j)} , \tag{9}$$

where $\underline{X} = \langle X_1, \cdots, X_M \rangle$

and we have

$$G(N) = \sum_{i=1}^{M} H(\underline{X}, i, M) F(X_i, N, R) . \tag{10}$$

Note also that Eq. (B) has now become

$$p(n_1, \cdots, n_M, n_{M+1}) = \left\{ \frac{1}{\beta(N-n)} \prod_{i=1}^{M} X_i^{n_i} \right\} G^{-1}(N) . \tag{11}$$

Utilization of the multiserver

We can find the utilization of the multiserver, U_{M+1}, by using Eqs. (11), (9) and (10) and summing, with $n_{M+1} = 0$. That is, the multiserver is not being utilized whenever there are zero people in the station. Thus,

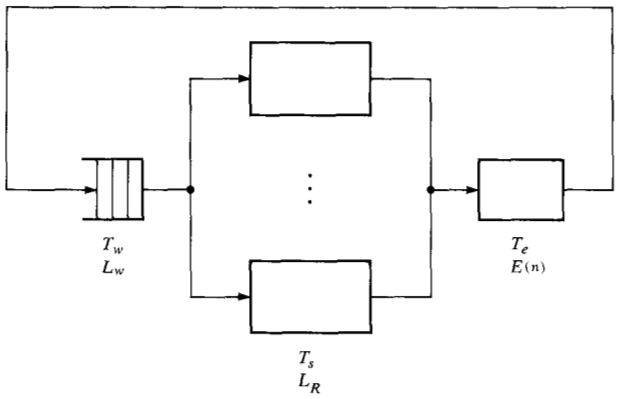
$$1 - U_{M+1} = \sum_{\substack{\langle n_1, \cdots, n_M \rangle \\ \sum_{i=1}^{M} n_i = N}} p(n_1, \cdots, n_M, 0) \tag{12}$$

$$= \sum_{\substack{\langle n_1, \cdots, n_M \rangle \\ \sum_{i=1}^{M} n_i = N}} \left\{ \prod_{i=1}^{M} X_i^{n_i} \right\} G^{-1}(N)$$

$$= \left(\sum_{i=1}^{M} H(\underline{X}, i, M) X_i^N \right) G^{-1}(N) . \tag{13}$$

Servers operating for the multiserver

We would like to find L_R, which is the expected number of servers operating at device $M+1$ (which is also the expected number of requestors being served). By definition:



L_w = length of waiting line
L_R = number of servers operating
$E(n)$ = number of requestors elsewhere
T_w = wait time
T_s = service time
T_e = elsewhere time

Figure 2 The network as seen by the multiserver.

$$L_R = \sum_{\langle n_1, \cdots, n_M \rangle} S(n)\, p(n_1, \cdots, n_M, n_{M+1}) \qquad (14)$$
$$\sum_{i=1}^{M} n_i \le N$$

where $S(n) = \begin{cases} N-n & N-n \le R \\ R & N-n \ge R \end{cases}$.

Using Eqs. (11), (9) and (10), we get:

$$L_R = \sum_{\langle n_1, \cdots, n_M \rangle} \frac{S(n)}{\beta(N-n)} \prod_{i=1}^{M} X_i^{n_i}\, G^{-1}(N) \qquad (15)$$
$$= \sum_{n=0}^{N} \frac{S(n)}{\beta(N-n)} \sum_{i=1}^{M} H(\underline{X}, i, M) X_i^{n}\, G^{-1}(N)$$
$$= \sum_{i=1}^{M} H(\underline{X}, i, M) F^{\Delta}(X_i, N, R)\, G^{-1}(N)\,, \qquad (16)$$

where

$$F^{\Delta}(X_i, N, R) = \sum_{n=0}^{N} \frac{S(n) X_i^{n}}{\beta(N-n)} \qquad (17)$$
$$= \frac{R}{R!} \sum_{n=0}^{N-R} \frac{X_i^{n}}{R^{N-n+R}} + \sum_{n=N-R+1}^{N} \frac{(N-n) X_i^{n}}{(N-n)!},$$

or reversing the order of summation:

$$F^{\Delta}(X_i, N, R) = \sum_{n=0}^{R} \frac{X_i^{(N-1)-n}}{n!} + \frac{1}{R!} \sum_{n=R+1}^{N-1} \frac{X_i^{(N-1)-n}}{R^{n-R}}$$
$$= F(X_i, N-1, R)\,. \qquad (18)$$

Substituting this in (16) we get:

$$L_R = \sum_{i=1}^{M} H(\underline{X}, i, M) F(X_i, N-1, R)\, G^{-1}(N) \qquad (19)$$

Details for the multiserver

Referring to Fig. 2, we can define the queuing network as seen by the multiserver. Note that we have lumped the average time spent not in the multiserver and the number of requestors not at the multiserver into single parameters. Now we use the proportion:

$$T_w : T_s : T_e : T_w + T_s + T_e = L_w : L_R : E(n) : N\,. \qquad (20)$$

We know N, T_s and have calculated L_R, and can find $E(n)$ from summing the queue lengths for all other queues. Therefore we can calculate the rest of the above variables, provided we can calculate the individual queue lengths for all other queues.

Utilization for the single servers

We can find the utilization for the single servers, U_i, as we did for the multiserver, by summing Eq. (11) with $n_i = 0$.

Thus we find $1 - U_i$ for $i = 1, \cdots, M$:

$$1 - U_i = \sum_{\langle n_1, \cdots, n_M \rangle} p(n_1, \cdots, n_{i-1}, 0, n_{i+1}, \cdots, n_M, n_{M+1})$$
$$\sum_{i=1}^{M} n_i \le N$$
$$= \sum_{\langle n_1, \cdots, n_M \rangle} \frac{1}{\beta(N-n)} \prod_{\substack{j=1 \\ j \ne i}}^{M} X_j^{n_j}\, G^{-1}(N)\,. \qquad (21)$$

Or letting:

$$H_i^*(\underline{X}, j, M) = H(\underline{X}, j, M) \frac{(X_j - X_i)}{X_j} \qquad (22)$$

and using Eqs. (9) and (10)

$$1 - U_i = \sum_{\substack{j=1 \\ j \ne i}}^{M} H_i^*(\underline{X}, j, M)\, F(X_j, N, R)\, G^{-1}(N) \qquad (23)$$

Queue lengths for the single servers

We would like to find L_{Qi}, the queue lengths (waiting line and in service) for the single servers. But

$$L_{Qi} = \sum_{\langle n_1, \cdots, n_M \rangle} n_i p(n_1, \cdots, n_M, n_{M+1})$$
$$\sum_{j=1}^{M} n_j \le N$$
$$= X_i \frac{\partial \sum_{\langle n_1, \cdots, n_M \rangle} p(n_1, \cdots, n_M, n_{M+1})}{\partial X_i}, \qquad (24)$$

since $p(n_1, \cdots, n_M, n_{M+1})$ is a product of $X_j^{n_j}$ in Eq. (B).

Now from Eq. (10)

$$\sum_{\langle n_1, \cdots, n_M \rangle} p(n_1, \cdots, n_M, n_{M+1})$$

$$= \sum_{j=1}^{M} H(\underline{X}, j, M)\ F\ (X_j, N, R)\ G^{-1}(N)\ .$$

Therefore

$$L_{Qi} = X_i \left(\sum_{j=1}^{M} H'(\underline{X}, j, M)\ F\ (X_j, N, R) + H(\underline{X}, i, M)\ F'\ (X_i, N, R) \right) G^{-1}(N)\ , \qquad (25)$$

where the prime denotes partial derivative with respect to X_i, or

$$L_{Qi} = \sum_{j=1}^{M} H(\underline{X}, j, M)\ A\ (i,j)\ F\ (X_j, N, R)\ G^{-1}(N) + (M-1)\ H\ (\underline{X}, i, M)\ F\ (X_j\ , N, R)\ G^{-1}(N) + H(\underline{X}, i, M)[N\ F(X_i, N, R) - F^*\ (X_i, N, R)]\ G^{-1}(N)\ , \qquad (26)$$

where $$A(i,j) = \begin{cases} \dfrac{X_i}{(X_j - X_i)} & j \neq i \\ \displaystyle\sum_{\substack{l=1 \\ l \neq i}}^{M} \frac{X_i}{(X_l - X_i)} & j = i\ . \end{cases} \qquad (27)$$

Since

$$F'(X_i, N, R) = \frac{N}{X_i} F(X_i, N, R) - \frac{1}{X_i} F^*(X_i, N, R)\ , \qquad (28)$$

$$F^*(X_i, N, R) = \sum_{n=0}^{R} \frac{nX_i^{N-n}}{n!} + \frac{1}{R!} \sum_{n=R+1}^{N} \frac{nX_i^{N-n}}{R^{n-R}}\ , \qquad (29)$$

and

$$H'(\underline{X}, j, M) = H(\underline{X}, j, M)\ \frac{1}{(X_j - X_i)} \qquad \text{when } j \neq i\ ,$$

$$H'(\underline{X}, i, M) = \frac{M-1}{X_i} H(\underline{X}, i, M) + H(\underline{X}, i, M) \sum_{\substack{l=1 \\ l \neq i}}^{M} \frac{1}{(X_l - X_i)}\ . \qquad (30)$$

Details for the single servers

We can find the rest of the statistics for the single servers, as we did for the multiserver, by using Eq. (20) where

$$L_{Qi} = L_{Wi} + L_{Ri}$$

and $L_{Ri} = U_i$ for single servers.

Since we know N, T_{si} and have calculated L_{Ri} and $L_{Wi} = L_{Qi} - L_{Ri}$, we can find all the others.

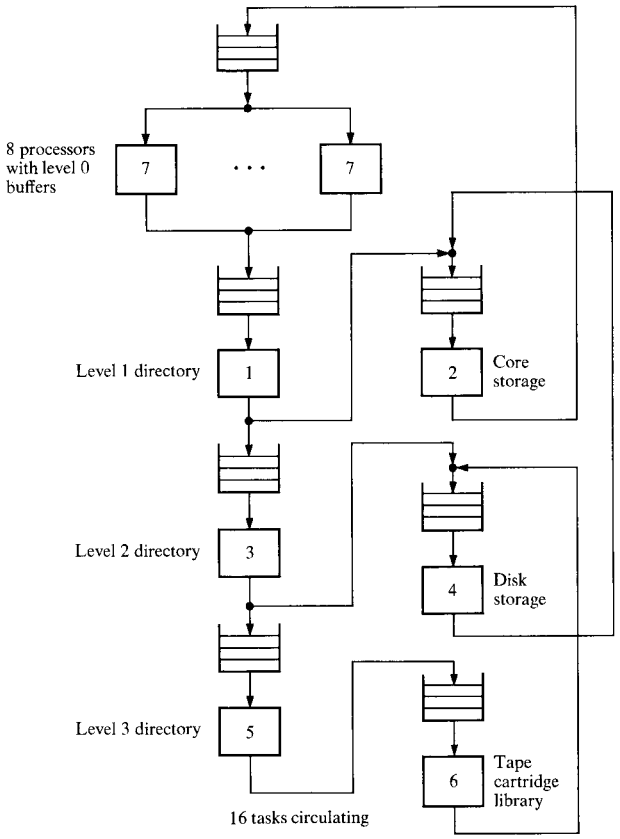


Figure A1 Typical multiprocessor system.

Other important statistics

One other statistic that can be calculated is the average minimum time for a requestor to return to the same queuing station once he leaves. This could be calculated by letting $N = 1$ in the above equations, or using Markov chain theory as in Ref. 4 since, if all waiting lines are empty, each station becomes a simple exponential delay.

Conclusions

In this paper we have simplified and extended Gordon and Newell's approach to closed queuing networks. In particular, we have greatly reduced the amount of computation involved and derived other statistics not derived by them.

Appendix

A typical multiprocessor computer system analyzed by the method given in this paper is shown in Fig. A1. The results of an analysis of this system are shown in Table A1.

Table A1 Results of a typical analysis

```
      INTERQUE
INTERCONNECTING QUEUE NETWORK WITH ONE MULTI SERVER QUEUE
      F. R. MOORE  10/7/70
STATION 7 IS THE MULTISERVER WITH 8 SERVERS
REQUESTORS-16
N     TRANSITION PROBABILITY MATRIX
0          0.99       0.01       0          0          0          0
0          0          0          0          0          0          1
0          0          0          0.99       0.01       0          0
0          1          0          0          0          0          0
0          0          0          0          0          1          0
0          0          0          1          0          0          0
1          0          0          0          0          0          0
```

SERVER	AVERAGE ELSEWHERE TIME	AVERAGE QUEUING TIME	AVERAGE WAITING TIME	SERVICE TIME	MINIMUM ELSEWHERE TIME
1	1402098.81	110.14	0.14	110.00	777520.00
2	1400162.30	2046.65	46.65	2000.00	775630.00
3	140218894.07	2000.46	0.46	2000.00	77761000.00
4	128723309.88	11497584.65	6497584.65	5000000.00	72763000.00
5	14017060797.16	5028655.53	28655.53	5000000.00	7771299999.99
6	13672638397.31	349451055.38	99451055.38	250000000.00	7526299999.99
7	152600.61	1249608.34	549608.34	700000.00	77630.00

SERVER	SERVER STATION UTILIZATION	AVERAGE QUEUE LENGTH	AVERAGE WAITING LINE LENGTH	AVERAGE SERVERS OPERATING	MINIMUM SERVERS OPERATING
1	0.0012552	0.001257	0.000002	0.001255	0.000141
2	0.0228211	0.023353	0.000532	0.022821	0.002572
3	0.0002282	0.000228	0.000000	0.000228	0.000026
4	0.5705284	1.311940	0.741411	0.570528	0.064298
5	0.0057053	0.005738	0.000033	0.005705	0.000643
6	0.2852642	0.398743	0.113479	0.285264	0.032149
7	0.9999998	14.258740	6.271343	7.987397	0.900171

```
RUNNING TIME-1.2  SECS.
      RUN DATE-2/29/72
```

References

1. W. J. Gordon and G. F. Newell, "Closed Queuing Systems with Exponential Servers," *Operations Research* **15**, 254–265 (1967).
2. T. L. Saaty, *Elements of Queueing Theory*, McGraw-Hill Book Co., Inc., New York, 1961, p. 330.
3. An explanation of generating functions is provided in *Introduction to Combinatorial Mathematics*, by C. L. Liu, McGraw-Hill Book Co., Inc., New York, 1968, although the book does not show this lemma.
4. A. L. Leiner, *Supermod: An Analytic Tool for Modeling the Performance of Large-Scale Systems*, RC 2796, IBM Corp., Feb. 12, 1970.

Received May 15, 1972

The author is located at the IBM Systems Development Division Laboratory, Poughkeepsie, New York 12601.