Proceedings of the 1986 Winter Simulation Conference
J. Wilson, J. Henriksen, S. Roberts (eds.)

INFINITE-SOURCE, AMPLE-SERVER CONTROL VARIATE MODELS FOR FINITE-SOURCE, FINITE-SERVER REPAIRABLE ITEM SYSTEMS

Mohamed A. Ahmed
Douglas R. Miller
Department of Operations Research
School of Engineering and Applied Science
The George Washington University
Washington, D.C. 20052

ABSTRACT

A new modeling idea for comparing infinite-source, ample-server models (∞/∞) and finite-source, finite-server models (f/f) is considered. This comparison provides an estimate of the error when approximating an f/f system with an ∞/∞ system, and allows analytical solutions of the ∞/∞ model to be used as control variates. This approach is applied to estimate the difference in performance between an M/G/∞ queueing system (∞/∞) and the classical machine repair problem (f/f) and the difference in performance between infinite-source, ample-server multiechelon repairable item inventory systems and finite-source, finite-server multiechelon systems. Using an ∞/∞ model as a control variate is shown to be an effective variance reduction technique for estimating the performance of many f/f systems.

1. INTRODUCTION

There is considerable interest in design and performance of repairable item systems. A simple case of a repairable item system is the machine repair model shown in Figure 1. The situation modeled has a population consisting of M items which we desire to be operational at all times and Y spares that support the system. There are C parallel repair channels. If more than C items require repair, a queue forms at the repair facility. Operating times until failure are exponentially distributed random variables with the mean time to failure of any item denoted by $1/\lambda$. Repair is generally distributed with mean time to repair denoted by $1/\mu$ (Cooper 1981, Gross and Harris 1985, Kleinrock 1975). A more complicated system is a multi-echelon repairable-item system as shown in Figure 2. Three bases and a depot are pictured. Items fail (independently of each other) after operating for an exponentially distributed length of time. There are repair shops at each base and the depot. Depending on the type of repair needed, some failed units must be sent to the depot to be repaired. Each repair shop has a certain number of repair channels. Repair times have a general distribution. Each base, as well as the depot, stocks spare units which, if available, are dispatched to the location from which the failed unit is received. If spares are not available, requests are backordered. Performance measures that we want to estimate include average numbers in or awaiting repair and availability of operational machines. We are interested in both transient and steady-state behavior of these systems.

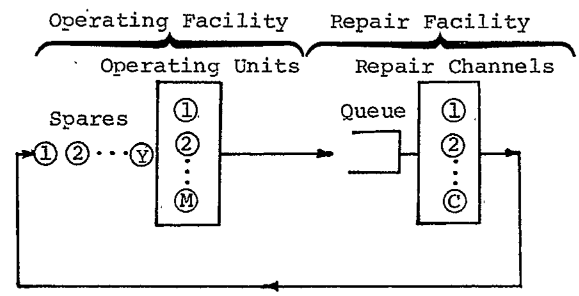

Figure 1: Schematic of a Machine Repair Model

These systems all have finite repair capacity and finite source (calling population). No analytical solution exists, except for some special cases with exponential operating times and exponential service times (Gross and Miller 1984; Gross, Miller, and Soland 1983, 1985).

There are models for the above types of repairable item inventory systems which make simplifying assumptions. In particular, it is

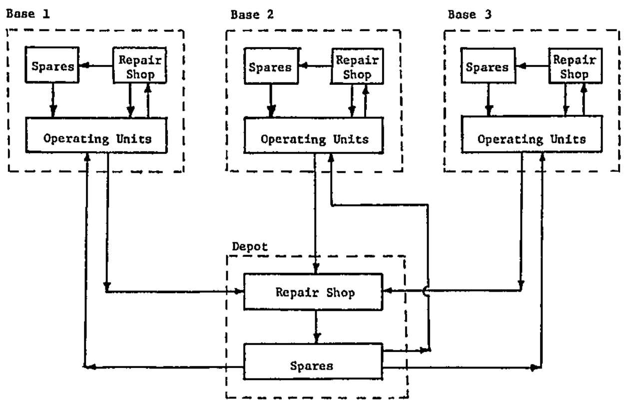

Figure 2: Three-base, One-depot Multiechelon Repairable Item System

assumed that the systems have ample repair facilities, i.e., no queue, and infinite source (calling population). Furthermore, these models do not address different backorder filling strategies. Models with simplifying assumptions for the machine repair system in Figure 1 could be an M/G/c queue (infinite source with failure rate $M\lambda$, finite repair capacity), or an $M/G/\infty$ queue (infinite source, ample server). Similar infinite-source, infinite-server models for the multiechelon repairable item system in Figure 2 are METRIC (Sherbrooke 1968) for steady-state analysis and Dyna-METRIC (Hillestad 1981) for transient analysis; both these models assume a Poisson calling population. These models are attractive because they can be solved efficiently with numerical algorithms.

Infinite-source, ample-server multiechelon models such as METRIC and Dyna-METRIC are much more tractable than finite-source, finite-repair-capacity models. However, they are only approximations for most multiechelon systems and hence they should only be used when they are "good" approximations, in which case they are clearly the model of choice. This leads us to the problem of computing or estimating the difference in performance between infinite-source, ample-server models (∞/∞) and finite-source, finite-server models (f/f). There are two main reasons for doing this: First, it will be a useful tool for determining whether the ∞/∞ model is an acceptable approximation; second, if it is not an acceptable approximation and we have an efficient estimation procedure for the difference, then the estimate can be used to correct the ∞/∞ solution. This second point amounts to using the ∞/∞ model as a "control variate" for the f/f model. We show that for many f/f systems it is computationally more efficient to compute the ∞/∞ solution using Dyna-METRIC, estimate the difference between f/f and ∞/∞, and add the two results rather than to estimate the behavior of f/f directly. Thus Dyna-METRIC can play a useful role as a control variate for more exact models. We believe similar results hold in steady state estimation using METRIC.

The purpose of this paper is to present a method for efficiently simulating the difference between the behavior of f/f and ∞/∞ models. The idea is to simulate a composite model that incorporates the behavior of both f/f and ∞/∞ models. We present this concept in Section 2 and illustrate it using the machine repair system of Figure 1. In Section 3 we show that this is an efficient way to estimate performance parameters of machine repair systems. In Section 4 we present a composite model for f/f and ∞/∞ models of the multiechelon system of Figure 2; and in Section 5 estimate the efficiency gained for some multiechelon test systems. Section 6 contains some concluding remarks.

2. A NEW MODELING IDEA: THE COMPOSITE MODEL

We introduce a composite model as a general idea to compare infinite-source, infinite-server models and finite-source, finite-server models. As an example, we present a special case comparing an $M/G/\infty$ queuing system (∞/∞) with the classical machine repair problem (f/f) which has exponential up-times and general service times. The difference between these two processes is estimated by simulating a more complicated open queuing network which is shown in Figure 3 and Table 1. This network includes both systems. A Poisson arrival process has rate $M\lambda$; this corresponds to arrivals to the ∞/∞ system and to the f/f system if all

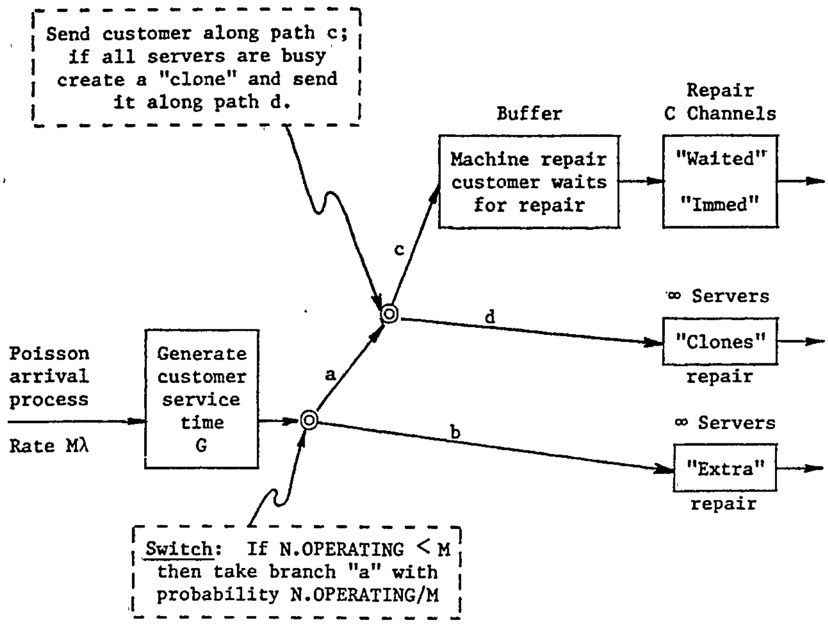

Figure 3: Network Whose Behavior Encompasses M/G/∞ Queue with Arrival Rate Mλ and Classical Machine Repair Problem with M Machines, Y Spares, C Repair Channels, Exponential (λ) Up-times, and General Repair Times.

M machines are operating. If fewer than M machines are operating there is an arrival rate lower than Mλ to the f/f system which is equivalent to a random thinning of the Poisson process. This thinning is state-dependent: It depends on the number of machines operating in the f/f system. In Figure 3, branch "a" corresponds to arrivals which exist for both systems (∞/∞ and f/f), and branch "b" corresponds to customers for the ∞/∞ system but not the f/f one; for example, if the number of operating machines equals M then customers take branch "a" with probability 1. The customer who took branch "a" also continues along path "c"; if such a customer does not wait in the buffer, he represents customers of both the ∞/∞ and the f/f systems and his service corresponds to service in the infinite-server repair shop and also service in the finite-server repair shop. However, if he has to wait in the buffer, then he represents only an arrival to the finite-server repair shop, and it is necessary to create a clone which receives immediate service by going along path "d" (infinite-server repair shop). Of the customers who went along path "c", we distinguish between those who were and those who were not cloned by labeling them "WAITED" and "IMMED", respectively. The state of this network is then given in terms of five variables: the number of each of four different customer types receiving service, and the number waiting in the buffer. From these five variables the number of customers in the f/f system and the number of customers in the ∞/∞ system are calculated. See Table 1.

The composite model simulates the common behavior of the two systems once and the special behavior for each system once. Figure 4 shows the overlap idea of the composite model as one of its advantages. In this model, if most of the machine-repair customers can enter repair without waiting in the buffer, i.e., if the system has something approaching ample service, then we get a significant overlap in Figure 4. This can be exploited to get variance reduction in estimation of the difference in performance between the f/f and ∞/∞ models.

3. COMPUTATIONAL EXPERIENCE WITH MACHINE REPAIR SYSTEM

We are interested in four different per-

Table 1: State Variables for Composite Machine Repair Model

N.EXTRA	= No. customers in system who took path b
N.CLONE	= No. customers in system who took path d
N.WAITED	= No. customers in repair who waited in buffer
N.IMMED	= No. customers in repair who started service immediately without waiting in buffer
N.BUFFER	= No. customers in buffer
N.F.REP	= No. customers in repair or awaiting repair for machine repair system
	= N.BUFFER + N.WAITED + N.IMMED
N.OPERATING	= No. of operating machines in the machine repair system
	= Min(M, M + Y - N.F.REP)
N.I.REP	= No. customers in M/G/∞ system
	= N.EXTRA + N.CLONE + N.IMMED
N.DELTA	= Difference between the machine repair system and M/G/∞
	= N.F.REP - N.I.REP
	= N.BUFFER + N.WAITED - N.EXTRA - N.CLONE

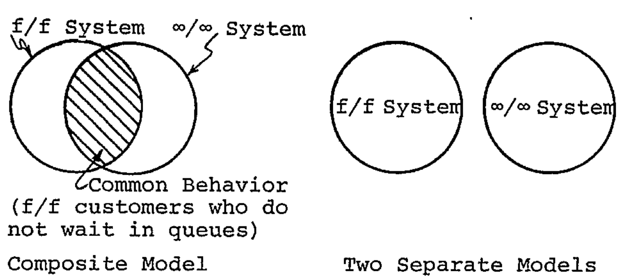

Figure 4: The Overlap Idea of the Composite Model

formance measures of the machine repair system. They are:

η_1 = Average no. items in or awaiting repair
η_2 = Average no. of busy repair channels
η_3 = Average no. of operating machines
η_4 = Probability that M machines are operating.

We estimate these performance measures for a transient system at time $t = 30$. We use two approaches. The first approach is to simply simulate an f/f model of the system. The second approach is to analytically compute the behavior of an ∞/∞ model of the system, simulate the difference in behavior of the f/f and ∞/∞ system, and add the estimates.

The efficiency measure of a procedure is the product of the variance of the estimator and the CPU time required to execute the procedure:

Efficiency = Variance * CPU time.

We estimate the efficiency by observing the CPU time for each execution and estimating the variance of the estimators of the performance measures of interest (Gross, Miller, and Plastiras 1984). Thus for the above two approaches we need to investigate the variance of the f/f model and the variance of the difference of the f/f model and the ∞/∞ model.

We wrote a SIMSCRIPT II.5 program to simulate the composite model of Figure 3. The classical event-scheduling method is used. The waiting customers are modeled as temporary entities. There are five types of events: arrival, immediate repair completion, waited repair completion, clone's repair completion, and extra repair completion. We observed difference of behavior between f/f and ∞/∞ using this program.

We also wrote a SIMSCRIPT II.5 program to simulate the f/f model of the machine repair system. The system was modelled as an open network similar to Figure 3, with a Poisson source which was thinned. There are two types of events: arrival and repair completion.

We used various cases as tests for comparing the different simulation approaches. Some of the cases considered are shown in Table 2. The repair times are Gamma with mean $1/\mu$ and shape parameter 2. These systems are initially in perfect condition with no failed machines. (One characterization of these systems is traffic intensity; in Table 2 this is given for the equivalent M/G/c system.) Each case was simulated for 1000 replicates. From our simulator of the composite model we estimated the difference between the f/f model and the ∞/∞ model for each of the performance measures η_i, $i = 1,2,3,4$; we also estimated the variances of these estimators. From our simulator of the f/f model we estimated the four performance measures for the f/f model as well as the variances of the estimators. We also observed the CPU times required to execute the simulations. The results for the test cases of Table 2 are given in Table 3. (Note that in Table 3 entries less than unity reflect superiority of the composite approach.) We see that the composite approach always requires more CPU time but that it always gives a variance reduction except in the case of η_2 for high traffic intensity. (This anomaly reflects the fact that the ∞/∞ model provides a horrible approximation for the distribution of busy channels for a system with high traffic intensity.) Overall the efficiency of the composite approach is superior to the straightforward approach of simulating the f/f system.

Table 2: Some Machine Repair Test Cases

Case #	M	Y	C	Traffic Intensity	λ	μ
1	24	6	4	.48	0.1	1.25
2	24	3	4	.48	0.1	1.25
3	24	0	4	.48	0.1	1.25
4	24	6	3	.64	0.1	1.25
5	24	3	3	.64	0.1	1.25
6	24	0	3	.64	0.1	1.25
7	24	6	2	.96	0.1	1.25
8	24	3	2	.96	0.1	1.25
9	24	0	2	.96	0.1	1.25

In order to estimate the performance of the f/f system using the composite simulator we must compute the behavior of the ∞/∞ system analytically. This can be done using basic properties of the transient M/G/∞ queue; see Gross and Harris 1985. Table 4 also gives 95% confidence intervals for the difference (DIF) in behavior estimated from the composite simulator, 95% confidence intervals for the f/f behavior obtained by adding the confidence interval for the difference to the analytic solution of the ∞/∞ model, and finally 95% confidence intervals for the f/f model obtained directly from the f/f simulator.

Comparing the composite approach and the straightforward simulation of the f/f model, we see that the composite approach produces a significant decrease in the variance of most of the estimators; but it also increases the CPU time. For light and medium traffic intensity, the composite model is much more efficient than direct simulation of the f/f model. Why is heavy traffic worse? This is because the difference between the f/f and ∞/∞ models is greater, which means they overlap less, and therefore ∞/∞ becomes a bad approximation of f/f. (See Figure 4.)

The composite model of Figure 3 is similar to the usual method of paired-comparison analysis: in effect, we are putting the same customer stream through two different systems (Law and Kelton 1982). What is different is that the composite model just performs calculations which are common to both systems once instead of twice. Thus we would expect only slight differences in the variances of the two approaches but significantly longer execution times for the traditional paired-comparison approach. We performed some calculations and found this to be true: the composite approach was between 25 and 50 percent more efficient than the paired-comparison approach for the nine test cases of Table 2.

4. A COMPOSITE MODEL FOR A MULTIECHELON SYSTEM

The multiechelon system in Figure 2 can also be analyzed using a composite model which encompasses both f/f and ∞/∞ models. Table 5 lists the system parameters and state variables for such a composite model. We model the depot and each of the three bases as open

Table 3: Comparisons of Composite Model and Finite/Finite Model of Machine Repair Systems

Case #	Variance η_1	Variance η_2	Variance η_3	Variance η_4	Average CPU Time	Efficiency η_1	Efficiency η_2	Efficiency η_3	Efficiency η_4
1	.058	.133	.500	.462	1.038	.060	.138	.519	.480
2	.055	.128	.156	.098	1.053	.058	.135	.164	.103
3	.107	.191	.107	.149	1.062	.114	.203	.114	.158
4	.327	.668	.853	.800	1.107	.362	.739	.944	.886
5	.279	.629	.496	.404	1.107	.309	.696	.549	.477
6	.237	.552	.237	.273	1.108	.263	.612	.263	.302
7	.873	8.408	.993	.991	1.186	1.035	9.972	1.178	1.175
8	.812	6.047	.895	.988	1.186	.963	7.172	1.061	1.172
9	.728	3.808	.728	.850	1.153	.839	4.390	.839	.980

(Ratios for Estimator of Performance)

Note: Entries correspond to Var(Composite)/Var(finite/finite), CPU(Composite)/CPU(finite/finite), and Efficiency(Composite)/Efficiency(finite/finite).

Table 4: Confidence Intervals for Performance Measures of Machine Repair Systems

Case #	∞/∞	DIF	F/F (∞/∞ + DIF)	F/F (Simulated)
		η_1 = Average Number of Items in Repair or Awaiting Repair		
1	1.916	.071 < .094 < .117	1.987 < 2.010 < 2.033	1.938 < 2.034 < 2.130
2	1.916	.029 < .051 < .073	1.945 < 1.967 < 1.989	1.899 < 1.991 < 2.083
3	1.916	-.141 < -.114 < -.087	1.775 < 1.802 < 1.829	1.742 < 1.826 < 1.910
4	1.916	.376 < .441 < .506	2.292 < 2.357 < 2.422	2.267 < 2.381 < 2.495
5	1.916	.289 < .345 < .401	2.205 < 2.261 < 2.317	2.179 < 2.285 < 2.391
6	1.916	.028 < .074 < .120	1.944 < 1.990 < 2.036	1.919 < 2.014 < 2.109
7	1.916	3.260 < 3.456 < 3.652	5.176 < 5.372 < 5.568	5.186 < 5.396 < 5.606
8	1.916	2.226 < 2.382 < 2.538	4.142 < 4.298 < 4.454	4.149 < 4.322 < 4.495
9	1.916	1.196 < 1.316 < 1.436	3.112 < 3.232 < 3.352	3.116 < 3.256 < 3.396
		η_2 = Average Number of Busy Repair Channels		
1	1.916	-.057 < -.028 < .001	1.859 < 1.888 < 1.917	1.834 < 1.912 < 1.990
2	1.916	-.078 < -.050 < -.022	1.838 < 1.866 < 1.894	1.812 < 1.890 < 1.968
3	1.916	-.203 < -.170 < -.137	1.713 < 1.746 < 1.779	1.695 < 1.770 < 1.845
4	1.916	-.089 < -.036 < .017	1.827 < 1.880 < 1.933	1.839 < 1.904 < 1.969
5	1.916	-.112 < -.061 < -.010	1.804 < 1.855 < 1.906	1.814 < 1.879 < 1.944
6	1.916	-.237 < -.189 < -.141	1.679 < 1.727 < 1.775	1.686 < 1.751 < 1.816
7	1.916	-.181 < -.098 < -.015	1.735 < 1.818 < 1.901	1.814 < 1.842 < 1.870
8	1.916	-.232 < -.152 < -.072	1.684 < 1.764 < 1.844	1.755 < 1.788 < 1.821
9	1.916	-.341 < -.265 < -.189	1.575 < 1.651 < 1.727	1.636 < 1.675 < 1.714
		η_3 = Average Number of Operating Machines		
1	23.986	-.021 < -.012 < -.003	23.965 < 23.974 < 23.983	23.968 < 23.980 < 23.992
2	23.798	-.044 < -.027 < -.010	23.754 < 23.771 < 23.788	23.714 < 23.757 < 23.800

Case #	∞/∞	DIF	F/F (∞/∞ + DIF)	F/F (Simulated)
3	22.081	.087 < .114 < .141	22.168 < 22.195 < 22.222	22.090 < 22.174 < 22.258
4	23.986	-.074 < -.053 < -.032	23.912 < 23.933 < 23.954	23.917 < 23.939 < 23.961
5	23.798	-.232 < -.190 < -.148	23.566 < 23.608 < 23.650	23.534 < 23.594 < 23.654
6	22.081	-.120 < -.074 < -.028	21.961 < 22.007 < 22.053	21.891 < 21.986 < 22.081
7	23.986	-1.235 < -1.113 < -.991	22.751 < 22.873 < 22.995	22.757 < 22.879 < 23.001
8	23.798	-1.742 < -1.609 < -1.476	22.056 < 22.189 < 22.322	22.034 < 22.175 < 21.316
9	22.081	-1.436 < -1.316 < -1.196	20.645 < 20.765 < 20.885	20.604 < 20.744 < 20.884
		η_4 = Probability M Machines Operating		
1	.996	-.011 < -.006 < -.001	.985 < .990 < .995	.980 < .987 < .994
2	.871	-.009 < -.002 < .005	.862 < .869 < .876	.836 < .858 < .880
3	.147	.007 < .016 < .025	.154 < .163 < .172	.136 < .159 < .182
4	.996	-.039 < -.029 < -.019	.957 < .967 < .977	.952 < .964 < .976
5	.871	-.086 < -.070 < -.054	.785 < .801 < .817	.765 < .790 < .815
6	.147	-.014 < -.003 < .008	.133 < .144 < .155	.118 < .140 < .162
7	.996	-.369 < -.340 < -.311	.627 < .656 < .685	.623 < .653 < .683
8	.871	-.451 < -.420 < -.389	.420 < .451 < .482	.409 < .440 < .471
9	.147	-.071 < -.055 < -.039	.076 < .092 < .108	.070 < .088 < .106

networks. The composite models of the bases are the same as in Figure 3 except the Poisson arrival process to Base i has rate $M_i \lambda_i \alpha_i$ instead of $M\lambda$, $i = 1,2,3$. The open network model of the depot is similar to that of the base except it has three Poisson arrival streams with rates $M_i \lambda_i (1-\alpha_i)$, $i = 1,2,3$, and it must also manage backorders and depot spares. The composite depot model is shown in Figure 5. When an item arrives at the depot, its repair time is generated and it is determined whether it is an arrival to the f/f system or only to the ∞/∞ system. Then it is necessary to either send a depot spare to the originating base or to create a backorder. The rest of the model is the same as Figure 3 except for the disposition of a repaired item: as an item leaves the network it, in effect, increments the depot spares pool or fills a backorder according to some backorder filling strategy.

5. COMPUTATIONAL EXPERIENCE WITH MULTIECHELON MODEL

We are interested in five different performance measures of the multiechelon repairable item system. They are:

ν_1 = Average no. items in or awaiting repair at base

ν_2 = Average no. items in or awaiting repair at depot

ν_3 = Average no. of operating machines at base

ν_4 = Probability that the desired machines are operating at a given base.

ν_5 = Probability that the desired machines are operating for the whole system.

To investigate the efficiency of the composite approach compared to straightforward simulation of the f/f model we proceed as we did for the machine repair system in Section 3. We consider nine test cases as described in Table 6. The repair times are Gamma with shape parameter 2. We simulated 500 replicates of each case for $0 \leq t \leq 30$; we then estimated the five performance measures and the variance of the estimators. The comparison of these variances for the composite approach and the straightforward f/f approach are given in Table 7. The ratios of CPU times and efficiencies are also given. We see as before that the composite model leads to smaller variances but longer execution times. Over this class of models and performance parameters there is a considerable improvement in effi-

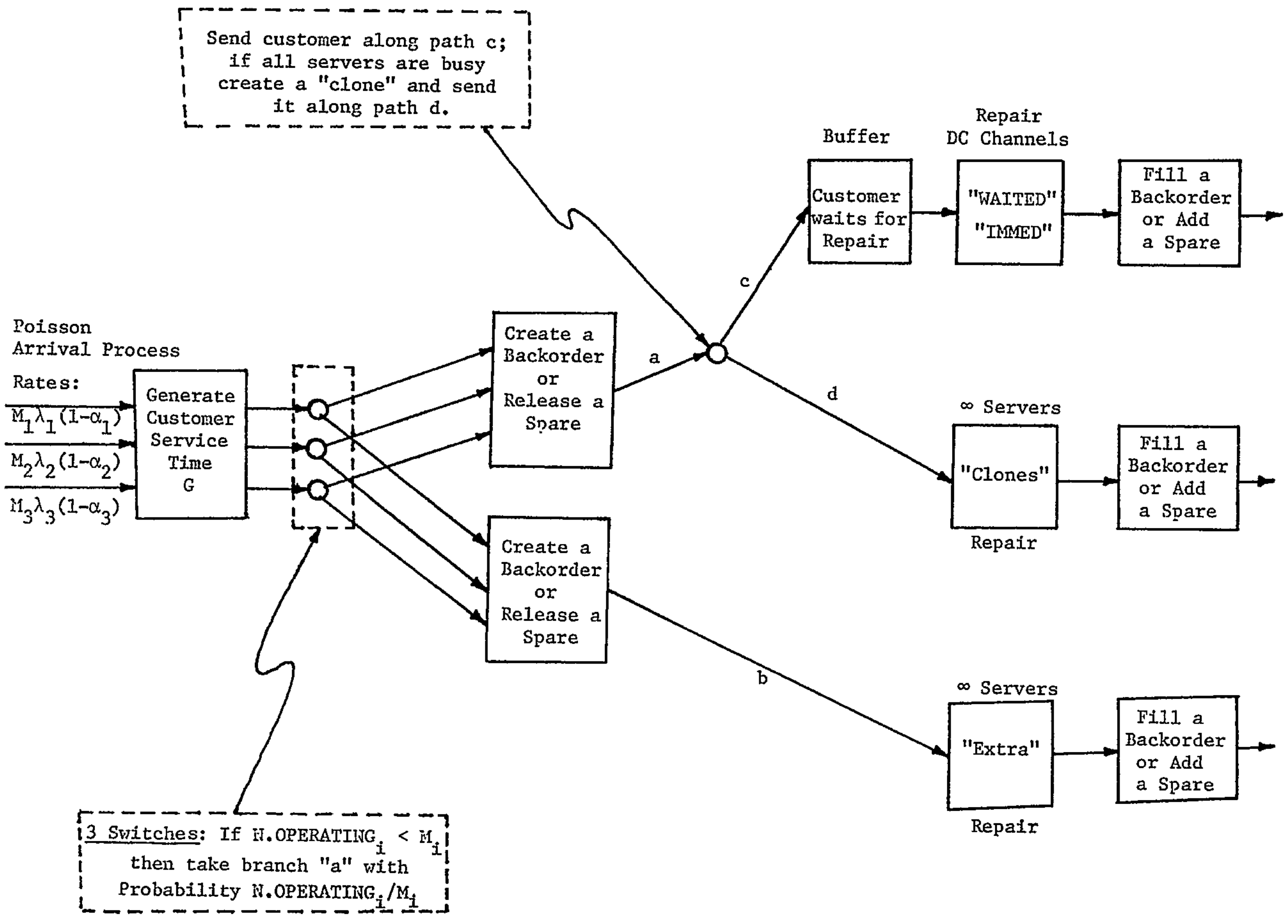

Figure 5: Open Network Whose Behavior Encompasses the Behavior of Depots of Two Three-base Multiechelon Systems: One Which is Infinite-source, Ample-server and the Other Which is Finite-source, Finite-server

ciency over the straightforward simulation of the f/f model.

6. CONCLUSIONS

This preliminary study shows that the composite model is an efficient way to estimate the difference in behavior between finite-source, finite-server models and infinite-source, ample-server models. We plan to use this method to study the behavior of multi-echelon repairable item inventory systems. There are important questions concerning the accuracy of METRIC and Dyna-METRIC, the effect of different backorder filling strategies, and optimal placement of spares and repair channels. It is important to simulate as efficiently as is reasonably possible in such a study.

ACKNOWLEDGMENT

The authors' work was supported by Office of Naval Research Contract N00014-86-K-0073.

Table 5: System Parameters and State Variables for Multiechelon Composite Model

λ(i)	= Failure rate, Base i items
μ(i)	= Repair rate, Base i items
M(i)	= Desired number of working machines at Base i
BC(i)	= Number of repair channels in repair shop at Base i
BS(i)	= Number of spares assigned to Base i
α(i)	= Probability that machine failing at Base i is base repairable
DC	= Number of depot repair channels
DS	= Number of depot spares
μ_D	= Depot repair rate
N.EXTRA(i)	= Number of customers in Base i who took path b
N.CLONE(i)	= Number of customers in Base i who took path d
N.WAITED(i)	= Number of customers in repair at Base i who waited in buffer
N.IMMED(i)	= Number of customers in repair at Base i who started service immediately without waiting in buffer
N.B.BUFFER(i)	= Number of customers in buffer at Base i
N.F.BACKORDER(i)	= Number of finite backorders required at Base i
N.I.BACKORDER(i)	= Number of infinite backorders required at Base i
N.OPERATING(i)	= Number of operating machines at Base i
	= Min(M(i), M(i) + BS(i) - (N.B.BUFFER(i) + N.WAITED(i) + N.IMMED(i) + + N.F.BACKORDER(i)))
D.S.FINITE	= Number of finite depot spares available
D.S.INFINITE	= Number of infinite depot spares available
N.D.EXTRA	= Number of customers in depot who took path b
N.D.CLONE	= Number of customers in depot who took path d
N.D.WAITED	= Number of customers in repair who waited in depot buffer
N.D.IMMED	= Number of customers in depot who started service immediately without waiting in depot buffer
N.D.BUFFER	= Number of customers in depot buffer

Table 6: Some Multiechelon Test Cases

Case #	M_i i=1,2,3	BS_i i=1,2,3	BC_i i=1,2,3	DS	DC	λ_i i=1,2,3	α_i i=1,2,3	μ_i i=1,2,3	μ_D
1	24	6	4	6	4	.1	.7	1.125	1.125
2	24	3	4	3	4	.1	.7	1.125	1.125
3	24	0	4	0	4	.1	.7	1.125	1.125
4	24	6	3	6	3	.1	.7	1.125	1.125
5	24	3	3	3	3	.1	.7	1.125	1.125
6	24	0	3	0	3	.1	.7	1.125	1.125
7	24	6	2	6	2	.1	.7	1.125	1.125
8	24	3	2	3	2	.1	.7	1.125	1.125
9	24	0	2	0	2	.1	.7	1.125	1.125

Table 7: Comparisons of Composite Models and f/f Models of Multiechelon System

Case #	Ratios for Estimator of Performance										
	Variance					Average CPU Time	Efficiency				
	ν_1	ν_2	ν_3	ν_4	ν_5		ν_1	ν_2	ν_3	ν_4	ν_5
1	.017	.040	*	*	*	1.095	.019	.044	1.095	1.095	1.095
2	.012	.039	.041	.000	.093	1.094	.013	.043	.045	.000	.102
3	.125	.089	.125	.000	.000	1.129	.141	.100	.141	.000	.000
4	.183	.348	1.000	*	1.000	1.119	.205	.389	1.119	1.119	1.119
5	.249	.326	.407	.272	.415	1.121	.279	.365	.456	.305	.465
6	.166	.241	.166	.069	.000	1.150	.191	.277	.191	.079	.000
7	.780	.878	1.000	1.000	1.000	1.164	.908	1.022	1.164	1.164	1.164
8	.770	.848	.871	.787	.971	1.166	.898	.989	1.016	.918	1.132
9	.609	.766	.609	.581	.500	1.197	.729	.917	.729	.695	.599

Note: * correspond to zero variance for both composite and f/f models.

REFERENCES

Cooper, R. B. (1981). Introduction to Queueing Theory, Second Edition. North Holland, Amsterdam.

Gross, D. and Harris, C. M. (1985). Fundamentals of Queueing Theory, Second Edition. John Wiley and Sons, Inc., New York.

Gross, D. and Miller, D. R. (1984). Multiechelon repairable item provisioning in a time-varying environment using the randomization technique. Naval Research Logistics Quarterly 31, 347-361.

Gross, D., Miller, D. R., and Plastiras, C. G. (1984). Simulation methodologies for transient Markov processes: A comparative study based on multi-echelon repairable item inventory systems. In: Proceedings of the Summer Computer Simulation Conference. Boston, 37-43.

Gross, D., Miller, D. R., and Soland, R. M. (1983). A closed queueing network model for multi-echelon repairable item provisioning. IIE Transactions 15, 344-352.

Gross, D., Miller, D. R., and Soland, R. M. (1985). On some common interests among reliability, inventory, and queueing. IEEE Transactions on Reliability R-34, 204-211.

Hillestad, R. J. (1981). Dyna-METRIC: Dynamic multi-echelon technique for recoverable item control. WD-911-AF, Rand Corporation, Santa Monica, California.

Law, A. M. and Kelton, W. D. (1982). Simulation Modeling and Analysis. McGraw-Hill, Hightstown, New Jersey.

Kleinrock, L. (1975). Queueing Systems, Vol. I. John Wiley and Sons, Inc., New York.

Sherbrooke, C. C. (1968). METRIC: A multi-echelon technique for recoverable item control. Operations Research 16, 122-141.

AUTHOPS' BIOGRAPHIES

MOHAMED A. AHMED is a Research Assistant in the Department of Operations Research at George Washington University. He has an M.S. in Operations Research from GWU. He is currently working on his D.Sc. in Operations Research.

Mohamed A. Ahmed
Department of Operations Research, SEAS
George Washington University
Washington, D.C. 20052, U.S.A.
(202) 676-6084

DOUGLAS R. MILLER is a Professor of Operations Research at George Washington University. He received his Ph.D. in Operations Research from Cornell, was Assistant and Associate Professor of Statistics at the University of Missouri-Columbia, and worked at NASA Langley Research Center before coming to GWU. He is a member of ORSA, ASA, and IEEE Computer Society. His research interests cover many topics within the realm of applied probability and statistics.

Douglas R. Miller
Department of Operations Research, SEAS
George Washington University
Washington, D.C. 20052, U.S.A.
(202) 676-7528