A Comparison of Transient Period Detection Algorithms Applied to Markovian Systems

M. W. McKinnon

Applied Research & Theoretical Sciences
P.O. Box 13164
Research Triangle Park, NC 27709
"b.mckinnon@ieee.org"

D. Tipper

University of Pittsburgh
Department of Information Science
Pittsburgh, PA 15260
"tipper@lis.pitt.edu"

ABSTRACT

The ramifications of statistical bias due to the initial state of a system are well known. Simply, long-term averages will be tainted by the observations which occur while the statistic approaches steady state. Determining realistic techniques for estimating the length of the transient period is a significant problem, particularly in systems with which there is no experience. This study focuses on the comparison and evaluation of several techniques for determining the length of transient periods. Markovian queues for which an exact length of the transient can be determined are used as the test-bed for these techniques.

KEYWORDS

Discrete Simulation, Queueing Models, Transient Length

INTRODUCTION

In using simulations of a stochastic system, the question of statistical bias due to system initialization has been significant in many works. Consider an arbitrary system in which some measure of the performance is of interest. It is likely that very little is known with regard to the steady state characteristics of such a system. In simulating such a system and initializing it to some arbitrary state, it is therefore not possible to know whether the starting state is typical of the steady-state behavior of the system. As such, the starting state may have an impact on long-term performance measures.

This point has motivated the study of algorithms which can determine the length of a transient period due to initialization bias, such that those observations which are collected during the transient period may be deleted. Other works in the literature (notably (Gafarian, et al. 1978), (Schruben 1982), (Schruben, et al. 1983), (Welch 1983), (Vassilacopoulos 1989), (Pawlikowski 1990), (Law and Kelton 1991), and (Roth and Josephy 1993)) also discuss the motivation for determining the initial transient length of a system simulation.

Several algorithms have been proposed for determining the length of transient periods; many are analytically justified while others are purely heuristic. A thorough survey of methods for determining transient period lengths is presented in (Gafarian, et al. 1978) and Section 3.1 of (Pawlikowski 1990). However, many of these algorithms are dependent on a diverse set of parameters and the guidelines for determining these parameters are usually conjectural.

In this paper, we implement and compare a number of techniques for estimating the transient period of a system. The techniques which we study are documented in (Gafarian, et al. 1978), (Pawlikowski 1990), and (Law and Kelton 1991). In this work, we seek to compare these techniques with one another as well as experimentally determine parameters which yield the most accurate estimates of the theoretically determined transient period (without significant overestimation). Note that the techniques described in (Gafarian, et al. 1978), however, are only compared in situations in which multiple replications are available. In many circumstances, repeating simulations and aggregating the results is not a feasible alternative or is computationally unattractive. In extending this work, we have applied these techniques to both single and multiple replications (as applicable).

Secondly, results for determining the length of the initial transient period that involve mean arrival rates for finite capacity systems which are greater than mean service rates have not been found in the literature. This situation, however, is common in studies of computer systems and communication networks (Lovegrove, et al. 1990).

The next section presents the background and parameters of the transient period detection algorithms associated with this work. In the following sections, the algorithms under consideration as well as references to their origins are presented along with the results of their implementation. A comparison of the results of these algorithms with theoretical values is also presented, followed by conclusions and recommendations.

BACKGROUND

Simulation models of systems are used to generate random samples on which statistical conclusions can be based. A basic problem of simulated systems is determining how long (either in terms of the number of observations or the time related to the observations) before "steady state" is reached. Typically, "steady state" is defined in terms of one of the statistics inherent to the system which eventually takes on certain properties or passes a pre-defined criteria. Most algorithms rely on the mean of a performance measure to determine steady state.

For this work, "steady state" is based on the mean sojourn time for a customer through the system under study. The system used for these experiments was a M/M/1/21 queue (20 buffer spaces and 1 server). As such, this system will allow for examining the time required to reach "steady-state" for overloaded systems. The arrival rate of customers to the system, λ, is varied from 0.1 to 2.0 in increments of 0.1 customers per second; the service rate, μ, is normalized to 1.0 customers per second. By numerically solving the Chapman-Kolmogorov equations describing the M/M/1/21 queue for varying levels of utilization and a 2% settling time, the exact transient period length may be obtained. These theoretical values are given in Table 1 for the denoted values of server utilization, $\rho = \lambda/\mu$.

This study involves comparing four transient period estimation techniques selected from (Gafarian, et al. 1978), (Pawlikowski 1990), and (Law and Kelton 1991). Each transient period estimation technique was compared in two different experimental ways. First we evaluate the techniques for their ability to determine the transient period from an experiment comprised of multiple replications with different pseudo-random number generator seeds. Secondly, the performance of each of the estimation techniques is also applied to a single (common) simulation replication, as applicable.

The performance measures for both of these situations are determined for two cases, where applicable: an "empty & idle" initial system state and a "known plausible steady state" initial system state. The latter is determined from the steady state estimate of the "empty & idle" single replication case. In the following sections, we discuss each of the four transient length estimation algorithms. Also presented are the numerical determinations of the transient periods under the aforementioned conditions.

"Crossings of the Mean" Rule

Referenced in (Gafarian, et al. 1978) and (Pawlikowski 1990), this algorithm is attributed to Fishman's Concepts and Methods in Discrete Event Digital Simulation (John Wiley, New York, 1973). The idea is that observations contribute to a running mean which is asymptotic to a statistic's theoretical mean. By the nature of a running mean, individual observations will appear on either side of the mean as the simulation progresses, contributing to its convergence.

$$\bar{X}(n) = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad (1)$$

Simply, as each observation is recorded by the simulation, the mean of the observations up to that point is calculated. The number of consecutive observation pairs which "cross" this running mean is then determined. A "cross" is defined as observation x_{i-1} falling either above or below the mean $\overline{X}(n)$ and observation x_i falling on the opposite side of $\overline{X}(n)$, respectively; $\overline{X}(n)$ is the sample mean given by equation (1). If the number of "crosses" is less than a previously determined threshold ("k"), the simulation continues; otherwise, steady state is deemed to have been realized.

TABLE 1: Theoretical Transient Periods

ρ	**Transient Period**	ρ	**Transient Period**
0.1	6.0	1.1	165.0
0.2	8.0	1.2	130.0
0.3	11.0	1.3	99.0
0.4	15.0	1.4	78.0
0.5	22.0	1.5	63.0
0.6	36.0	1.6	52.0
0.7	63.0	1.7	44.0
0.8	109.0	1.8	39.0
0.9	162.0	1.9	34.0
1.0	184.0	2.0	30.0

The main disadvantages of this approach are twofold. One is that it requires the storage of all observations, x_i, until steady state is determined since the number of crosses must be re-evaluated with every new observation contributing to the running mean. Secondly, there is not a general method for determining an appropriate value for the parameter 'k'.

In the experiments presented here, the number of crosses allowed was varied from 5 to 20. The first set of experiments were performed with an "empty and idle" start-up condition: the system was empty and the server idle when the simulation was begun. Figure 1 illustrates the results obtained with a lone replication for various values of k as compared to the theoretical values presented in Table 1. Figure 2 shows the similar results for 25 replications per experiment for various values of 'k'. In previous studies of this type of algorithm, (Wilson and Pritsker 1978b) suggested that k = 7 would be sufficient for a M/M/1/15 system with 1000 observations. (Gafarian, et al. 1978) showed that k = 25 was sufficient for an M/M/1 system with 100 replications of 100 ensemble observations each.

Figure 1 shows that for the single replication case, transient periods corresponding to lower utilizations (ρ<0.6) were best estimated using low values of k. Even at k = 5, transient period estimate errors do not drop under 100% until ρ=0.5 (7.05% error). As the server utilization increases, higher values of k provide more accurate results. Note that for utilizations around one (ρ=1.0), the length of the transient can be underestimated unless 'k' is increased.

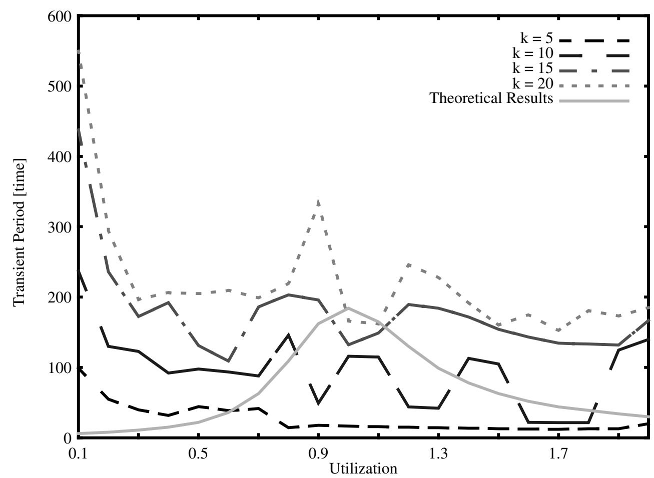

FIGURE 1

By implementing 25 replications, Figure 2 bears out similar results. Again, the curves significantly exaggerate the transient periods for lower utilizations. This figure shows that for ρ>1.0, the curves for the respective values of 'k' do not closely track the theoretical results, but they do follow the shape of the theoretical results' curve.

Once the first set of simulations reached "steady state" according to the "Crossings of the Mean" rule, the number of entities in the system was recorded. (In the multiple replication case, the average of the number of entities in each replication was rounded to the next higher integer.) This value was then used as the initial state for another set of similar experiments, the results for which are given in (McKinnon and Tipper 1995).

The effects of changing the initial state of the system seemed to reduce the transient period length somewhat except for utilizations below 1.0. The effect of changing the initial state for utilizations above 1.0 did not seem particularly evident as the values for the length of the initial transient estimates were about the same as when the simulation started in the “empty & idle” state.

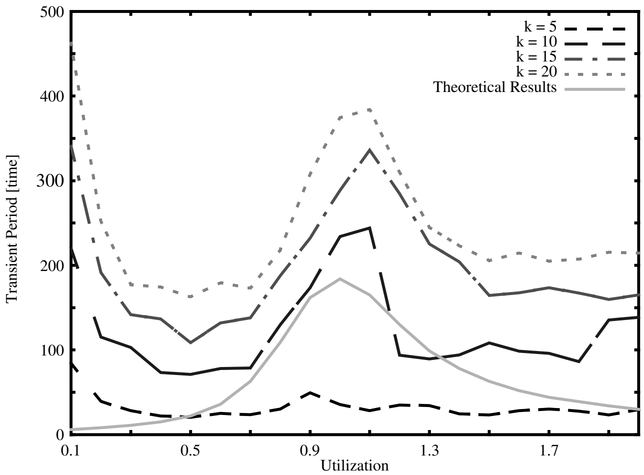

FIGURE 2

“Window of Tolerance” Rule

Described in (Pawlikowski 1990), this algorithm is based on evaluating the relative volatility of the running mean over a pre-defined number of samples. It basically states that a fixed number of consecutive running mean values must lie within a “tolerance zone” in terms of a percentage of a given running mean. Therefore, the transient period has completed with observation ‘n’ once the criteria in Equation (2) is met for a predetermined number of observations (‘k’).

$$\frac{\left|\bar{X}(n+k) - \bar{X}(n+i)\right|}{\left|\bar{X}(n+k)\right|} < \delta \qquad (2)$$

$$\forall i \in \{1 \ldots k\}$$

One obvious problem with this technique is the fact that two separate and unrelated parameters must be defined before any analysis can commence: the length of the “window” of observations (‘k’) and the acceptable deviation from the steady state value within the “window” (‘δ’). From an implementation standpoint, additional complexity is introduced in that the ‘k’ previous observations must be constantly maintained and updated.

After running the same simulation (seeded with the same random number stream) from the “Crossings of the Mean” Rule with this criteria, the results in Figure 3 & Figure 4 were generated. The experiments were performed for a number of sets of parameters. Based on the suggested values given in (Pawlikowski 1990), the length of the “window”, ‘k’, (i.e., the number of samples required to fall within the pre-specified tolerance level) was varied from 10 to 25 in increments of 5 and from 25 to 100 in increments of 25. The reason that values below 30 were considered and are presented are described in (Pawlikowski 1990), (Gafarian, et al. 1978), and (Roth 1985): overestimation of the transient period. These values are denoted ‘k’ in all figures’ legends. The tolerance values (denoted ‘δ’) were assigned values from 0.1% to 5%. The more stringent requirements, those below 1%, did not reliably generate a termination point for the length of the transient period within the 100,000 event simulations used for this study. As such they are not considered in the following discussion. Similarly, the values of ‘k’ shown are those which most closely estimated the theoretical transient period; values of ‘k’ below 10 were not significantly different from those shown and those above 15 generated dramatically larger transient period estimations.

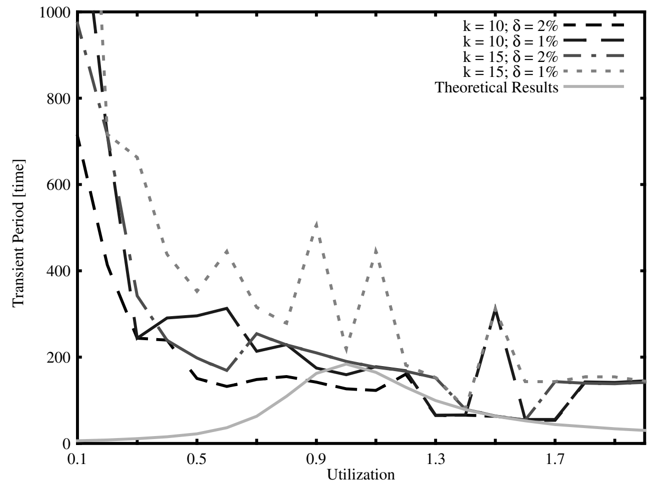

FIGURE 3

Figure 3 illustrates the values generated for the “single replication” case; it illustrates that, for low utilizations (ρ<0.6) in this case, all of the shown sets of parameters grossly overestimated the length of the transient period. The curves comprised of the aggregate values of 25 replications are shown in Figure 4 and were somewhat more indicative of the effects of each of the parameters. All of the transient period lengths estimated for low utilizations were at least an order of magnitude greater than that of the theoretical values. (Roth 1985) and (Gafarian, et al. 1978) both mention that this type of heuristic can result in extreme over-estimation of transient periods. As the utilization increased to ρ>0.9, most of the values of ‘k’ and ‘δ’ shown underestimate the length of the transient period. Also, none of the curves shown resemble that of the theoretical values.

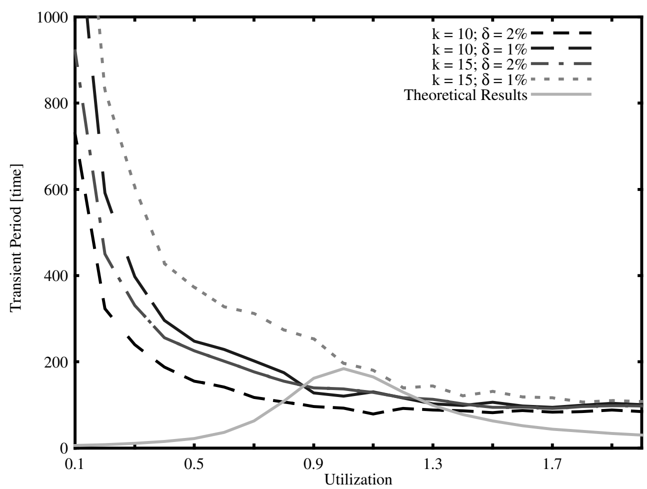

FIGURE 4

By re-initializing the simulation with the “steady state” average queue length from the previous experiment, results similar to those documented in the previous experiments are obtained. Again, this technique has been considered within the literature as probably inappropriate for this type of scenario.

"WELCH'S ALGORITHM"

Described in (Law and Kelton 1991) & (Welch 1983), this algorithm is the only one analyzed which specifically requires multiple replications of identical systems. In this method, 'n' replications, each of length 'm', are generated, from which an ensemble average across the 'n' replications for each observation is determined. Therefore, given observation 'i' (of 'm' observations) from the replication j, $x_{i,j}$, equation (3) describes the ensemble averages, 'X_i'.

$$X_i = \frac{1}{n}\sum_{j=1}^{n} x_{i,j} \tag{3}$$

A fixed number, 'w', of consecutive ensemble averages are then averaged with one another in order to attempt to remove high-frequency artifacts. 'Y_i' is defined as the "smoothed" average value of the observations.

One modification for this scheme provides for multiplying each of the ensemble averages by carefully selected coefficients, a_l, before averaging them. (Law and Kelton 1991) suggests that the application of a low-pass filter to the ensemble average values can improve the estimate of the transient response. One set of appropriate coefficients and smoothed average values are shown in equations (4), (5), and (6).

$$Y_i = \sum_{l=-(i-1)}^{i-1} a_l X_{i+l} \tag{4}$$

$$i \in \{1, \ldots, w\}$$

$$Y_i = \sum_{l=-w}^{w} a_l X_{i+l} \tag{5}$$

$$i \in \{w+1, \ldots, m-w\}$$

$$a_l = \left[\frac{(1+\cos(l\pi/h))}{2h}\right] \tag{6}$$

The subscript 'l' ranges from 1 to 'w', the odd (non-even) size of the "window" of observations which contribute to the ensemble averages. A plot of the smoothed averaged observations 'Y_i' must then be generated and a point beyond the apparent transient period must be selected by visual observation by the practitioner.

Regarding the selection of parameters for "Welch's Algorithm", (Law and Kelton 1991) and (Welch 1983) present general guidelines for 'n' & 'w'. Additionally, the length of each replication, 'm', is a third parameter which must be determined prior to execution of the simulation. All of the simulations in this study had 'm' set to 10,000 events.

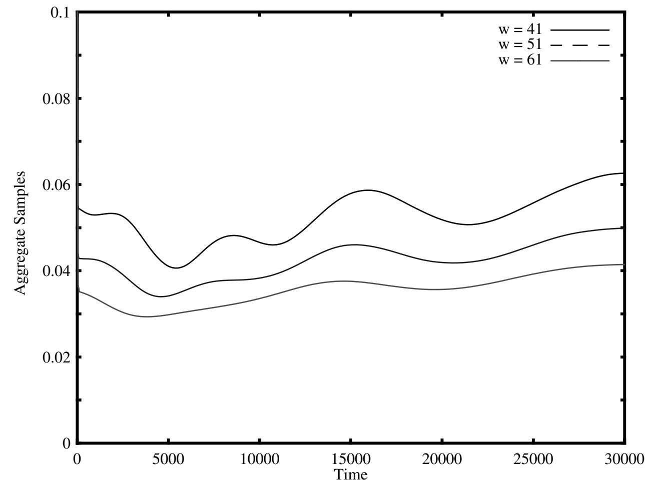

FIGURE 5

(Law and Kelton 1991) made two points regarding "Welch's Algorithm" which were verified in this work. First, increasing 'w', the window of ensemble averages which contribute to each "aggregate" statistic, results in a more stable "aggregate" statistic curve. This point is illustrated in Figure 5, using ρ=0.6 and n=25 as an example, the size of the "window" was increased from 41 to 61.

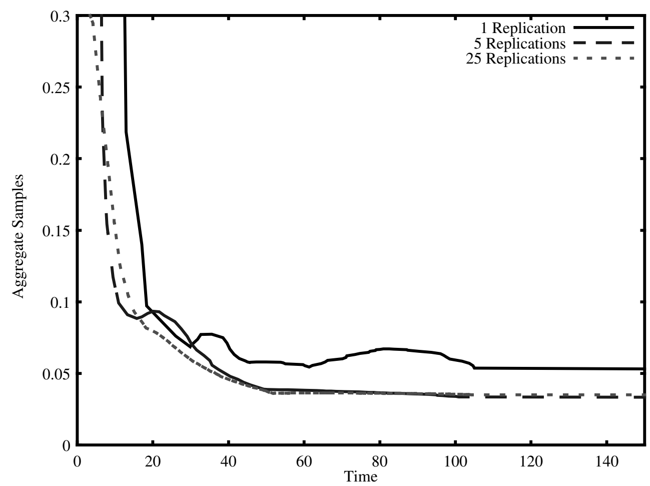

FIGURE 6

Second, by increasing 'n' (the number of replications contributing to each ensemble average), the volatility of the "aggregate" statistic was dramatically affected. For instance, again using ρ=0.6 and a window size of w=51 samples for comparison, Figure 6 shows how dramatically the number of replications affected the variation of the "aggregate" statistic

Since "Welch's Algorithm" calls for visually estimating when steady-state has begun, some error is intrinsically associated with results derived from this algorithm. These errors will be directly linked to the resolution of plots based on experiments, which are based on the number of observations and the number of replications in each run. (Law and Kelton 1991) calls for a minimum of 2000 observations (‘m’) and 10 replications (‘n’) per scenario with w between 10 and half of the number of observations.

Figure 6 shows a comparison of “aggregate” statistics between results from one, five, and 25 replications. Since “Welch’s Algorithm” specifically calls for multiple replications, the value of the single replication data may be minimal.

With the previous notes in mind, the interpretations of the results for 25 replications per experiment are shown in Figure 7. It should be noted that, based on visual observations, Welch’s Algorithm significantly underestimated the transient period length for ρ between 0.7 and 1.5 for all values of ‘w’ considered.

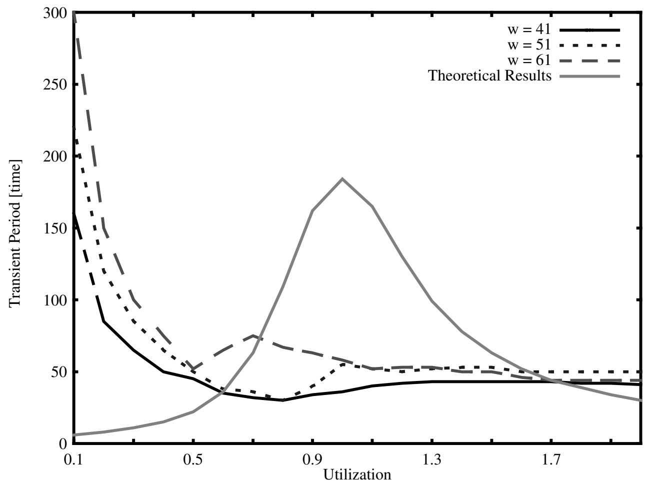

FIGURE 7

Using the ensemble averages of the queue lengths of the concurrent replications at the time steady state is deemed to have occurred as an initial state, the model was re-simulated. The estimated transient period under this condition is approximately halved across all utilizations, hence further underestimating the theoretical transient period length for ρ>0.7.

Schruben’s “Test for Stationarity”

The “Test for Stationarity” Rule is described briefly in (Pawlikowski 1990) and, in detail, in (Schruben et al. 1983). Using one of the heuristic rules in the literature as a starting point for the transient period, the transient period’s observations are removed until the remaining observations pass a statistical stationarity test. Given the number of observations removed, additional observations are generated at the end of the event stream. The ‘T’ measure in equation (7) is then compared to the theoretical Student’s ‘t’ distribution statistic (for a predefined significance and number of degrees of freedom). The process is repeated until the criteria is satisfied by the experimental measure ‘T’ passing a statistical test for stationarity.

$$T = \frac{\sqrt{45}}{n_t^{1.5}\sqrt{n_v}\,\sigma_{n_v}\left(n_t - n_v\right)} \times \sum_{k=1}^{n_t}\left(\bar{X}_{n_0}\left(n_t\right) - \bar{X}_{n_0}(k)\right)\left(k - \frac{k^2}{n_t}\right) \tag{7}$$

This algorithm obviously requires a comparatively large number of variables to be considered before execution. (Pawlikowski 1990) provides recommended values based on a separate algorithm which incorporates that of (Schruben, et al. 1983). The four of these parameters which relate to the previous equation and their recommended values are presented in Table 2.

In the implementation of this test, the “Crossings of the Mean” Rule was used as a starting point. Since Schruben’s rule only allows for increasing the alleged transient, it will not be able to compensate for a starting algorithm’s overestimation.

TABLE 2: “Test for Stationarity” Rule Parameters

Variable	Recommended Value
Simulation Length (n_{max})	100000
Sequence Length Used to Estimate Stationarity (n_t)	200 (minimum)
Sequence Length Used to Estimate Steady-State Variance, σ_n (n_v)	100
Exchange Rate between Transient Observations & New Observations	0.5 (New Observations per Transient Observation)

As we presented earlier, the “Crossings of the Mean” Rule substantially overestimated the transient period for most utilizations, thereby somewhat defeating this technique. The results for the “Test for Stationarity” based on the “Crossings of the Mean” Rule (for a single replication and an “empty & idle” initial state) are presented in Figure 8. The comparable results for the “possible steady state” initial state are presented in (McKinnon and Tipper 1995).

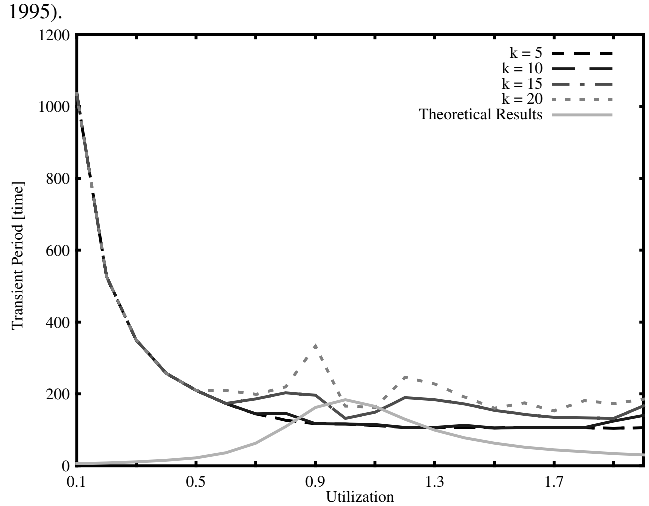

FIGURE 8

Another note regarding not only Schruben’s test, but also many other tests, is the requirement of a minimum number of observations in order to determine if a transient has been found. Realizing at least 100 (n_v) observations are necessary to accurately compute σ over n_v, Schruben’s test determined that in many low utilization cases the transient period had (legitimately) expired after the required 100 observations. This phenomena is what is observed in both Figure 8 at lower utilizations (ρ<0.7) and for both of the entire k=5 curves. In other cases, however, specifically those about ρ=1.0, the test was able to improve the initial estimate

CONCLUSIONS

Based on the theoretical results, the transient length detection techniques' performance can be characterized as a function of the utilization of the queue. The "Crossings of the Mean" Rule was the most accurate at low utilizations for single replication experiments, although it significantly overestimated the actual transient period. In light of the number of observations which should be available for this type of experiment, however, the additional observations may be able to be sacrificed. It also performed best when given a reasonably "stable" starting state (i.e., a system state close to that of "steady state"); with ρ<0.3, transient estimates were reduced to roughly 50% of the estimates realized for an "empty & idle" system. Above that point, estimates were reduced slightly, but not appreciably. The appropriate size of the threshold value used for this type of experiment seemed to shift with the utilization, first appealing to lower thresholds and then, as the utilization increased, higher thresholds.

For multiple replications and low utilizations, Welch's Algorithm generated the most accurate results. The main hurdle associated with using it, however, is the necessity to manually estimate when the transient period has ended. This point makes it unreasonable to use in automated studies unless this requirement is satisfied using another algorithm (such as the "Crossings of the Mean" Rule or Rule "R8" from (Pawlikowski 1990) applied to the "aggregate" statistics). Applying an initial non-empty state did not affect the transient period estimates significantly.

As the utilization grows into the heavy utilization range (above roughly 75% of capacity), almost all of the algorithms discussed underestimated the transient period's length. The "Crossings of the Mean" Rule showed the results closest to the theoretical estimates in the single replication case and came very close to the theoretical results in the multiple replication cases (while overestimating the theoretical results). The effects of providing a plausible "steady state" as an initial condition to these experiments were negligible. The fact that the curves for higher values of 'k' (of the "Crossings of the Mean" Rule) did seem to approximate the shape of the theoretical curve in many cases does appear promising though.

The "Window of Tolerance" Rule provided the most accurate results for finite queue utilizations above approximately 1.3. At some set of parameters, the "Window of Tolerance" Rule overestimated the length of the transient period, but, as seen in Figure 3 and Figure 4, the higher values of 'δ' and lower values of 'k' provided the most accurate results. As the utilization approached 2.0, however, estimates still showed errors approaching 100%. Providing a non-empty initial state seemed to have a minimal effect on the estimate of the transient period in this case.

It should be noted that none of the transient length detection algorithms was found to accurately estimate the transient period over all load ranges. Furthermore, the parameters of each algorithm which yielded the most accurate estimation varied with the load. Based on the results of our study, we could suggest the "Crossings of the Mean" Rule with the use of multiple replications to estimate the transient period. If only a single replication is available, again, the "Crossings of the Mean" Rule would appear to yield the best results with appropriate selection of the parameter 'k'.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Krzysztof Pawlikowski of University of Canterbury, Christchurch, New Zealand for his suggestions.

REFERENCES

Gafarian, A. V., C. J. Ancker, and T. Morisaku 1978. "Evaluation of Commonly Used Rules for Detecting 'Steady State' in Computer Simulations." Naval Research Logistics Quarterly 78: 511-529

Law, A. M. and W. D. Kelton. 1991. Simulation, Modeling, and Analysis. McGraw-Hill, NewYork

Lovegrove, W. P., J. L. Hammond, and D. Tipper. 1990. "Simulation Methods for Studying Nonstationary Behavior of Computer Networks", IEEE Journal on Selected Areas in Communications 8. no. 9 (Dec.): 1696-1708

McKinnon, M. W and D. Tipper. 1995. "A Comparison of Transient Period Detection Algorithms Applied to Markovian Queueing Systems". Technical Report. Applied Research & Theoretical Sciences, RTP, NC.

Pawlikowski, K. 1990. "Steady-State Simulation of Queueing Processes: A Survey of Problems and Solutions." ACM Computing Surveys 22. no. 2 (June): 123-170

Roth, E. 1985. "A Relaxation Time Heuristic for Reducing Initialization Bias in Simulation of $M/E_k/1$ Queueing Systems." In Proceedings of 1985 Winter Simulation Conference, ACM/IEEE, New York, 263-268

Roth, E. and N. Josephy. 1995. "Relaxation Time Heuristic for Exponential-Erlang Queueing Systems." Computers & Operations Research 20. no. 3 (Apr.): 293-301

Schruben, L. W. 1982. "Detecting Initialization Bias in Simulation Output." Operations Research 30. no. 3 (May-June): 569-590

Schruben, L., H. Singh, and L. Tierney. 1983. "Optimal Tests for Initialization Bias in Simulation Output." Operations Research 31. no. 6 (Nov.-Dec.): 1167-1178

Vassilacopoulos, G. 1989. "Testing for Initialization Bias in Simulation Output", Simulation 52. no. 4 (Apr.):151-153

Welch, P. D. 1983."Statistical Analysis of Simulation Results." In Computer Performance Modelling Handbook, S. S. Lavenberg, ed. Academic Press, New York, 267-329

Wilson, J. R. and A. A. B. Pritsker. 1978. "A Survey of Research on the Simulation Startup Problem." Simulation 31. no. 3 (Sept.): 55-58

Wilson, J. R. and A. A. B. Pritsker. 1978. "Evaluation of Startup Policies in Simulation Experiments." Simulation 31. no. 3 (Sept.): 79-89

BIOGRAPHY

Martin W. McKinnon received a B.E.E. from Georgia Tech in 1989 and the Master's degree in Computer Engineering from Clemson University in 1993. He has been employed by The MITRE Corporation and Bell-Northern Research. In 1994, he joined Applied Research & Theoretical Sciences as a Scientist. He is currently completing his Ph.D. in Computer Engineering at N. C. State University. His current interests include transient queueing analysis, queueing models of ATM & optical networks and perturbation analysis. He is a member of IEEE, ACM, and ORSA.

David Tipper is an Associate Professor in the Department of Information Science at the University of Pittsburgh. He received the Ph.D. in Electrical Engineering from the University of Arizona in 1988.