Heuristic Rules for Improving Quality of Results from Sequential Stochastic Discrete-Event Simulation

KRZYSZTOF PAWLIKOWSKI, DONALD MCNICKLE and JONG-SUK R. LEE

University of Canterbury, Christchurch, New Zealand

Abstract

Sequential analysis of output data during stochastic discrete-event simulation is a very effective practical way of controlling statistical errors of final simulation results. Such stochastic sequential simulation evolves along a sequence of consecutive checkpoints at which the accuracy of estimates, usually conveniently measured by the relative statistical error, defined as the ratio of the half-width of a given confidence interval (at an assumed confidence level) to the point estimate, is assessed. The simulation is stopped when the error reaches a satisfactorily low value.

One of problems with this simulation scenario is that the inherently random nature of the output data produced during a stochastic simulation can lead to accidental, temporary satisfaction of the stopping rule. Such premature stoppings of simulations is one of causes of inaccurate final results, producing biased point estimates, with confidence intervals that do not contain the exact theoretical values.

In this paper we consider a number of rules of thumb that can enhance the quality of the results from sequential stochastic simulation despite that some simulations can be prematurely stopped. The effectiveness of these rules of thumb is quantitatively assessed on the basis of experimental results obtained from fully automated simulations aimed at estimation of steady-state mean values.

Keywords: Coverage of confidence intervals, sequential stopping rules, statistical errors of results, stochastic discrete-event simulation

1. INTRODUCTION

Stochastic discrete-event simulation is the most commonly used method in performance evaluation studies of such complex systems as, for example, modern telecommunication networks [1]. Since any such study should be regarded as a simulated statistical experiment, error bounds are essential to ensure credibility of the final results. A very effective practical way of controlling statistical errors of the simulation results is to analyze the errors sequentially during the simulation. This is known as the sequential scenario of stochastic discrete-event simulation or, simply, sequential simulation. Its practical significance is supported by arguments of the estimation theory: traditional fixed-sample size sampling is unable to provide a bounded-width confidence interval of a point estimator if its variance is unknown [2], but this problem does not exist when one applies sequential sampling; see e.g. [**?**].

Sequential stochastic discrete-event simulation evolves along a sequence of consecutive checkpoints at which the accuracy of estimates is assessed. This accuracy is usually conveniently measured by the relative statistical error, defined as the ratio of the half-width of the estimated confidence interval (CI), at an assumed confidence level, and the point estimate of the performance measure of interest. The simulation is stopped when this error assumes a satisfactorily low value.

Locations of consecutive checkpoints should be carefully considered. While the number of new observations (i.e. new items of output data) collected between two checkpoints should be sufficiently large to potentially cause a noticeable change in the level of error of estimates, too large distance between checkpoints can unnecessary make simulation longer. Too short distance between checkpoints can unnecessarily intensify processing of output data.

In this paper we restrict our attention to simulation experiments based on long single runs, which is the typical approach used when studying steady-state performance of stochastic dynamic systems. For example, if steady-state mean value μ of a performance measure is estimated at a consecutive checkpoint, one estimates μ by $\bar{X}(n) = (1/n)\sum_{i=1}^{n} x_i$, where $x_1, x_2, ..., x_n$ represent n observations $x_1, x_2, ..., x_n$ which had been collected before reaching that checkpoint. The relative error of the mean at that point is measured by

$$\epsilon(n) = \frac{\Delta(n)}{\bar{X}(n)}, \tag{1}$$

where $\Delta(n) = t_{df,1-\alpha/2}\hat{\sigma}[\bar{X}(n)]$ is the current half-width of the CI for μ at an assumed $(1-\alpha)$ confidence level, $0 < \alpha < 1$, $t_{df,1-\alpha/2}$ is $(1-\alpha/2)$-quantile of Student T random variable with df degrees of freedom, and $\hat{\sigma}[\bar{X}(n)]$ is the estimate of standard deviation of $\bar{X}(n)$. The simulation will stop at a given checkpoint iff

$$\epsilon(n) \leq \epsilon_{max}, \tag{2}$$

where ϵ_{max} $(0 < \epsilon_{max} < 1)$ is the largest acceptable relative error of the final results at the $(1-\alpha)$ confidence level. Otherwise, the simulation continues, more observations are collected, and the error is analyzed again, when the next checkpoint is reached. The advantage of using such relative measure of statistical errors is that simulators do not need to know the magnitude of performance measures they want to analyze. The asymptotic validity of such a stopping rule of sequential simulation has been proved in [3].

Observations collected during a single simulation run are correlated, so finding a good estimator of the standard deviation of $\bar{X}(n)$, $\sigma[\bar{X}(n)]$, is a major problem. More than a dozen of methods have been proposed for its estimation. A survey of sequential methods of mean value analysis proposed until 1990 can be found in [4]. Newer methods, or improved versions of previously proposed ones, can be found, e.g. [5], [6], [7], [8], [9] and [10]. Further in this paper we support our theses by two methods of mean value analysis: an unsophisticated Non-Overlapping Batch Means (NOBM) in its sequential version formulated in [4], and a more elaborated method of Spectral Analysis (SA), originally proposed in [11], in its sequential version modified in [4] and enhanced in [12] and [13]. More detailed specification of these methods are given in Section 2. All methods of steady-state output analysis involve different approximations and the only way of assessing their quality is by an exhaustive analysis of coverage of the confidence intervals they produce, i.e. analysis of the frequency with which the final confidence intervals from replicated simulations contain the true (theoretical) mean value of a given performance measure. A given method of analysis of simulation output data is regarded as correct if the confidence interval of the experimental coverage contains the theoretical confidence level assumed for the analyzed performance measure. A methodology for accurate, sequential analysis of coverage has been formulated in [14].

A robust sequential method of output data analysis can be automated and incorporated in user-friendly simulation tools, such as for example Arena, a general purpose commercial simulator [15], or Akaroa2, a universal controller of sequential simulation [16]. The main idea behind a fully automated sequential simulation is that it can be executed with a limited knowledge of the simulated processes and output data analysis is conducted on-line, during the simulation.

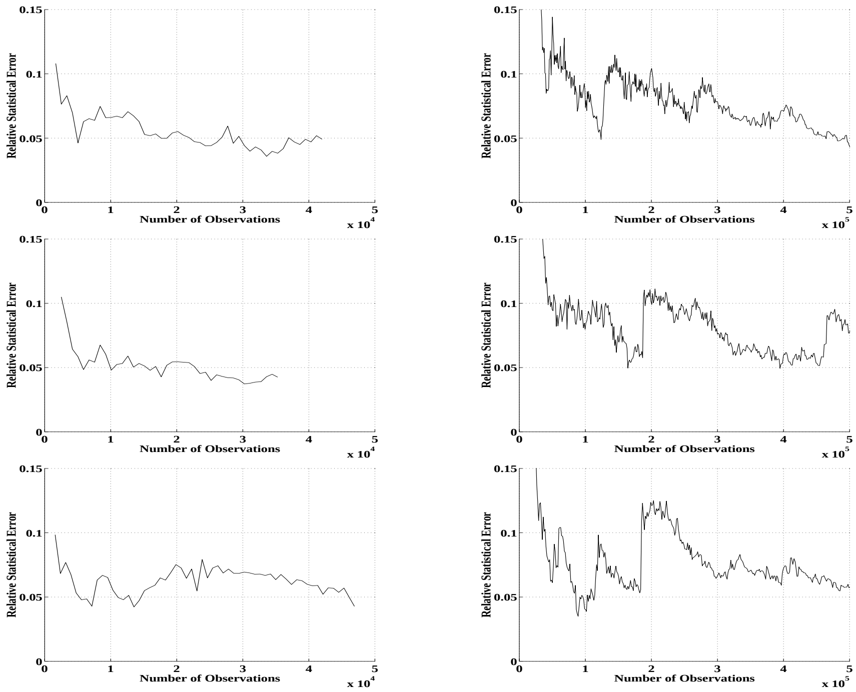
The practical significance of sequential stochastic simulation has been recognized for example in [17]. As one can read there, "... no procedure in which the run length is fixed before the simulation begins can be relied upon to produce a confidence interval that covers <the true theoretical value> with the desired probability ...". However, implementations of this idea should address a specific problem of such simulation, related with the fact that *inherently random nature of simulation output data can cause an accidental, temporal satisfaction of the assumed stopping rule of the simulation.* This can lead to very inaccurate final results: biased point estimates with confidence intervals that do not contain exact theoretical values.

Typically, in a long simulation run, the convergence of the relative statistical error of Equation (1) to its threshold value can be slow, but persistent, as shown in Figure 1. One can see that in some cases, during initial stages of simulation, the value of the error can prematurely reach low level and stay there for a longer time. In automated setting of sequential simulation, it would be associated with early stopping of simulation. Such prematurely finished simulations can produce very inaccurate results.

Experimental evidence of this phenomenon and the resulting significant degradation of the coverage of the final results of the simulation are documented in Section 2. In Section 3 we consider simple heuristic rules that can protect a sequential simulation against its premature stopping. Their effectiveness is quantitatively assessed in Section 4. This is followed by Conclusions in Section 5.

2. EXPERIMENTAL EVIDENCE

In this paper we consider heuristic rules which could be used for improving accuracy of any sequential method of output data analysis in steady state simulation. Without loosing generality of our discussion, we restrict ourselves



(a) load = 0.5 (b) load = 0.9

Figure 1: Convergence on a relative statistical error of Equation (1) in the case of the sequential version of Spectral Analysis (SA) when estimating steady-state mean response time in the $M/M/1/\infty$ queueing system at load equal 0.5 and 0.9.

to two methods of analysis of steady-state means (and their standard deviations): (i) Non-Overlapping Batch Means, and (ii) Spectral Analysis, in their sequential implementations.

Non-Overlapping Batch Means (NOBM) is an unsophisticated sequential version of the method of Batch Means, formulated in [4].

Spectral Analysis (SA) is a more elaborated method of output data analysis which explicitly takes into account existing correlations between observations by estimating the variance of the steady-state mean from the value of averaged periodogram at the frequency equaled zero.

Sequential versions of both methods have been implemented using uniformly spaced checkpoints. It is the most appropriate spacing between checkpoints if one intends to speedup simulation by means of Multiple Replications in Parallel; see [18].

The reason for using these versions of NOBM and SA as test cases for our rules of thumb is that their coverage has been thoroughly tested [**?**], and they have been used by users of two popular open-source simulation packages for modelling telecommunication networks, NS2 and OMNET++, when their simulations are executed under control of Akaroa2; see [19] for appropriate software interface and documentation.

All reported simulation results were obtained using a good quality pseudo-random number generator: a composite linear congruential generator proposed in [20], with the cycle longer than 10^{57}. Such a cycle has allowed to avoid accidental introduction of undesirable correlations in simulation output data which could happen if the cycle were exhausted and numbers were repeated. Additionally, to avoid overlapping of sequences of pseudo-random numbers used in different replications, each next replication of simulation used the last pseudo-random number of the previous replication as its first pseudo-random number.

For studying the consequences of prematurely finished simulation runs one needs to know the exact values of analysed performance measures. Having in mind applications of simulation in performance evaluation studies of telecommunicaton networks, we used three analytically tractable queueing systems: $M/D/1/\infty$, $M/M/1/\infty$ and $M/H_2/1/\infty$, with the coefficient of variation of service times ranging from zero ($M/D/1/\infty$) to 10 ($M/H_2/1/\infty$) Figure 2 gives histograms of the run-lengths of 10000 independent simulation replications, when estimating the mean response time in the corresponding queueing system at load of 0.9, with a relative statistical error $\epsilon_{max} = 0.1$

at a confidence level of 0.95. The simulation run-lengths for both methods, NOBM and SA/HW, were measured by the number of collected observations at which a given sequential simulation has stopped. The empirical mean run-lengths over 10000 sequential steady-state simulations are presented in Table 1.

The theoretically expected values of these simulation run lengths can be calculated using methods based on [**?**]. These are: 137921 observations (in the case of $M/M/1/\infty$), 55255 observations (in the case of $M/D/1/\infty$), and 536553 observations (in the case of $M/H_2/1/\infty$). These values are represented by vertical broken lines in Figure 2. Comparing recorded run-lengths of the simulation in Figure 2 with the expected number of observations required, we can see that many runs do not yield enough observations. Further, the average experimental run-length is always substantially shorter than that required theoretically.

The simulation run-lengths under NOBM and SA/HW are presented in Tables 2-4. Each of the results was obtained from 10000 independent replications of the sequential steady-state simulation. Following the proposal in [14], we have classified a simulation as 'too short' if its run-length was shorter than a threshold, taken to be one standard deviation below the mean simulation run-length. The threshold values of the minimum acceptable run-lengths of simulations and the overall experimental mean simulation run-lengths are given in the last two columns. The second and fourth columns give, respectively, the absolute and the relative number of 'too short' simulation runs over all 10000 replications of simulations, executed at each load level of each queueing system.

The quality of the final results produced by the 'too short' simulation runs can be assessed by their coverage, i.e. by the experimental frequency with which the final CIs of the results contains the theoretical value of the estimated parameter. In an ideal situation, the coverage should be close to the assumed confidence level. However, a closer look at the results from

	$M/M/1/\infty$	$M/D/1/\infty$	$M/H_2/1/\infty$
Sequential analysis Using NOBM	80,967	39,129	281,427
Sequential analysis Using SA/HW	106,037	44,845	403,492

Table 1: Mean run-lengths over 10000 replications of sequential steady70-state simulations (estimation of the mean response time at load $\rho = 0.9$, maximum relative statistical error of 10% at confidence level = 0.95)

the 'too short' simulation runs reveals that the coverage of the CIs of these simulation results can be very poor indeed; see the third column in Tables 2-4.

On the other hand, results of our coverage analysis of NOBM and SA/HW, obtained by applying the principles of sequential coverage analysis formulated in [14], show that these methods are able to offer the final results of similar quality (in the sense of coverage) if the 'too short' runs are eliminated. Figure 3 demonstrates this phenomenon. It shows the convergence of coverage for sequential steady-state simulations for $M/M/1/\infty$ queueing system at the traffic intensity $\rho = 0.8$. A jump in the current value of coverage, clearly seen in each of these figures is associated with discarding of all results obtained from 'too short' simulation runs.

These results show how wrong final simulation results obtained from 'too short' simulation runs can be in practice. Such a problem needs to be recognised in practical applications of fully automated sequential steady-state simulations. Rules for elimination of results that are responsible for poor coverage are discussed in the next section.

3. HEURISTIC RULES FOR IMPROVING THE QUALITY OF THE FINAL RESULTS

Most sequential simulations are run *only once* until the acceptable level of statistical error is reached. However, as illustrated by the experimental results in Section 2, a single sequential simulation run can be 'too short', leading to erroneous results, regardless of the output data analysis method used [21], [22]. Our results show that the problem becomes more critical when dealing with heavily loaded queueing systems, or, equivalently, with processes with stronger autocorrelations. The question is how to eliminate such results of poor quality.

One obvious solution is to repeat a sequential simulation several times and accept only results of good (or better) quality. This is not a new idea. As D. Knuth wrote in 1969 *"... the most prudent policy for a person to follow is to run each Monte Carlo program at least twice, using quite different sources of pseudo-random numbers, before taking the answers of the program seriously"* [23].

In this section, we propose five simple 'rules of thumb' which could help to eliminate results of poor quality, together with their motivations. Those rules are based on two ideas: (i) using only one run of several executed runs

(Rules I to III), or (ii) using all runs without discarding any results (Rules IV and V).

3.1. Heuristic Rules: I

Since we have evidence that the coverage of the final results is reduced by too short simulation runs, a simple rule of thumb can be formulated as follows:

1. Execute R independent replications of a given simulation and record the run-lengths (measured by the size of the sample of simulation output data).
2. Accept the result produced by the longest simulation run only.

Using the results in Tables 2 - 4, we can construct an upper bound for the probability that, having applied Rule I, one would still end up with the final results coming from a 'too short' simulation run. Namely, if one executes R independent replications, $R \geq 1$, and P_{short} is the probability that a simulation run is 'too short', then $(P_{short})^R$ is the probability that all R independent replications will belong to the class of 'too short' simulation runs. Using the worst cases from the sequential steady-state simulations presented in Section 2, the probabilities of 'too short' runs are given in Table 5. The probability quickly becomes negligible with an increased number of runs.

3.2. Heuristic Rules: II

The relative statistical error randomly changes with the number of collected simulation observations, although in the long run it reduces with the number of observations. The smaller the relative statistical error, the better the accuracy of the final results. Thus, one way of producing the most accurate final result could be to take it from the simulation run that has finished with the smallest relative statistical error. This gives us the following rule:

1. Execute R independent replications of a given simulation and record the final relative statistical error of the result.
2. Accept the result with the smallest relative statistical error only.

3.3. Heuristic Rules: III

Wider CIs have better chance of containing the theoretical value and thus should be characterised by better coverage than narrow CIs. Thus, one way to improve coverage of final results of sequential steady-state simulations could be to take the results which produced the widest CIs, by applying the following rule:

1. Execute R independent replications of a given simulation and record the final CIs of the results.
2. Accept the result with the widest CI only.

3.4. Heuristic Rules: IV

To reduce the randomness of results obtained from sequential steady-state simulations, one can combine a number of results obtained from independent replications of a given simulation. For example, one could propose the following rule:

1. Execute R independent replications of a given simulation, and record the run-lengths (measured by the size of the sample of simulation output data) and the estimated values.
2. Produce the final CI by combining all results obtained from R independent replications.

Rule IV needs a mean μ and a variance σ^2 of a combined simulation run to construct the combined CI. The mean value of a combined simulation should be calculated by weighting the results from R simulation runs, which have different mean values, calculated over different sample sizes. The variance of the combined mean can be calculated by using the usual unbiased estimator of variance for pooled samples.

Suppose one has variance estimates $s_1^2, s_2^2, \cdots, s_R^2$, from R independent samples of size $n_1, n_2, \cdots, n_R$, from populations with a common variance σ^2. Then, the pooled sample variance is calculated by

$$s_p^2 = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2 + \cdots + (n_R - 1)s_R^2}{(n_1 - 1) + (n_2 - 1) + \cdots + (n_R - 1)}, \tag{3}$$

which is an unbiased estimator of the variance σ^2.

3.5. Heuristic Rules: V

Trying to achieve a satisfactory level of coverage with acceptable statistical errors of the final results from a sufficiently long sequential simulation, one can simply combine ideas behind Rules III and IV, and thus propose the following rule:

1. Execute R independent replications of a given simulation.
2. Record the run-lengths (measured by the size of the sample of simulation output data) and the final CIs of the results.
3. Calculate the mean value as averaged over results from R independent replications.
4. Accept a half-width of a CI from a simulation run with the widest CI among R independent replications (Rule III).
5. Construct a CI with the results obtained in steps 3 and 4.

3.6. Summary

The proposed rules are a significant diversion from the common concept of running an automated sequential simulation only once, without even a pilot run [24], [4]. Note that Rules I to III discard (R - 1) replications and use only one replication to calculate the final results, while Rules IV and V use results from all R independent replications. It may seem strange to consider methods that discard runs in this way, however we point out that (i) the cost of simulation continues to decrease, and (ii) we have found that practical users of simulation are notoriously unprepared to carry out even conventional statistical analysis of their results [1]. This suggests that methods which simply involve discarding runs have considerably greater chance of acceptance and use than those which require additional post-processing. Of course, no heuristic rule of thumb can ensure that the final CIs from a stochastic simulation will contain the theoretical value, with a probability equal to the assumed confidence level. However, these heuristic rules may help reduce the use of results of poor quality. In the next section we compare the effectiveness of these rules of thumb when they are applied to our reference queueing models.

4. COMPARISONS OF THE PROPOSED HEURISTIC RULES

In this section, we study the effect of Rules I to V on the quality of the final results, in terms of coverage of CIs when applying the two methods of output data analysis (NOBM and SA/HW) in the case of $M/M/1/\infty$, $M/D/1/\infty$, and $M/H_2/1/\infty$ queueing systems. The mean response time was estimated with $\epsilon_{max} \cdot 100\% = 10\%$ as the upper level of the acceptable relative statistical error of the final results, at a confidence level of 0.95. In each case the final results are averaged over 2,000 independent replications. For example, in the case of $R = 5$ replications, we have used results from a total of 10,000 replications.

4.1. Heuristic Rule I

Figure 4 shows the application of Rule I, which uses the longest run of the executed R replications; for $R = 1$, 2, 3 and 5. The coverage of the final results clearly shows that Rule I is viable, and the larger R is, the better the quality of the final results.

In all cases considered, there is no need to assume that R is larger than 3, since the resulting coverage reaches a satisfactory level at this point. This is because as the statistical data of Table 5(a) shows, the probability that the remaining replication is still 'too short', after discarding two shorter replications out of three, drops to 0.006 or less even for the worst case; the SA/HW method in the $M/M/1/\infty$ queueing system.

4.2. Heuristic Rule II

The results of the coverage obtained by applying Rule II, which takes the most 'accurate' result, i.e., the result with the smallest relative statistical errors, out of R executed replications ($R = 1$, 2, 3 and 5) are depicted in Figure 5. From these, one can see that discarding the results with larger (but still acceptable) level of the final relative error actually *worsens* the coverage, regardless of the number of executed replications: a larger R will make the resulting coverage even worse. This is because the simulation producing the apparently most accurate results, in terms of the relative statistical error, has the narrowest CIs. These narrow CIs may sometimes be caused by the sudden (temporary) drop of the required level of relative statistical error, resulting in accidental stopping of simulation with an insufficient number of observations. Consequently, Rule II should not be applied.

4.3. Heuristic Rule III

The consequence of application of Rule III, which takes the result with the widest CIs of R replications; $R = 1$, 2, 3 and 5, is shown in Figure 6. As we can see, taking the simulation results with wider CIs improves the coverage of the final results, for each R, $R > 1$. However, the improvements of coverage are poorer than in the case of Rule I, especially when applied to the simulation of heavier loaded queueing systems; see Figure 4.

4.4. Heuristic Rule IV

Figure 7 shows the consequences of applying Rule IV: combining R replications for $R = 1$, 2, 3 and 5. Not surprisingly the larger the number of replications executed, the better the coverage and also the better (i.e. narrower) the CIs obtained. Generally speaking, as in the case of Rule I and III, there is no need to use R larger than 3, since the coverage obtained by combining $R = 3$ replications is already above the required level of confidence, 0.95. In all cases considered, combining R independent replications together always guaranteed that final results are produced with a (very) high coverage, since the final results are always produced from a large sample of observations.

4.5. Heuristic Rule V

The results of the coverage when applying Rule V (a combination of Rules III and IV), are depicted in Figure 8. The results are similar to those obtained by applying Rule IV. In general, however, Rule V produces a slightly higher coverage than Rule IV. Therefore, this rule of thumb is more desirable than Rule IV, if one always wants to obtain the final results with the highest coverage.

5. CONCLUSIONS

The proposed heuristic rules for improving the quality of the final results from sequential simulations have been analysed experimentally by applying them to the two methods of simulation output data analysis: NOBM and SA/HW, in simulation studies of $M/M/1/\infty$, $M/D/1/\infty$, and $M/H_2/1/\infty$ queueing systems. The results clearly show that Rules I, IV and V are viable in practice, since they ensure that credible final results are obtained with an acceptable coverage as the number of replications, R, increases. However, the results show that there is no need to repeat sequential simulation more than $R = 3$ times. While (not surprisingly) the more complex Rules IV and V, requiring the combination of results, (and hence, of course, the effective doubling or tripling of the number of observations) turn out to give the best results, the much simpler and easier to implement Rule I also does well. The proposed rules can be easily implemented in simulation packages, offering automated control of the relative statistical error of the final results in a sequential steady-state simulation.

[1] K. Pawlikowski, H. J. Jeong, J. R. Lee, On credibility of simulation studies of telecommunication networks, IEEE Communications Magazine 40 (1) (2002) 132–139.

[2] E. L. Lehmann, Notes on the Theory of Estimation, University of California Press, Berkeley, 1951.

[3] M. Ghosh, N. Mukhopadhyay, P. K. Sen, Sequential Estimation, Wiley and Sons, N.Y., 1997.

[4] P. W. Glynn, W. Whitt, Asymptotic validity of sequential stopping rules for stochastic simulations, Annals of Applied Probability 2 (1992) 180–198.

[5] K. Pawlikowski, Steady-state simulation of queueing processes: A survey of problems and solutions, ACM Computing Surveys 2 (1990) 123–170.

[6] D. Goldsman, K. Kang, Cramer-von mises variance estimators for simulations, in: Proceedings 1991 Winter Simulation Conference, WSC'91, IEEE Press, Phoenix, Arizona, 1991, pp. 916–920.

[7] R. B. Howard, M. A. Gallagher, K. W. Bauer, P. S. Maybeck, Confidence intervals for univariate discrete-event simulation output using the kalman filter, in: Proceedings 1992 Winter Simulation Conference, IEEE Press, Phoenix, Arizona, 1992, pp. 586–593.

[8] G. S. Fishman, Monte Carlo: Concepts, Algorithms and Applications, Springer-Verlag, N. Y., 1996.

[9] E. J. Chen, W. D. Kelton, A stopping procedure based on phi-mixing conditions, in: Proceedings 2000 Winter Simulation Conference, IEEE Press, San Diego, 2000.

[10] N. M. Steiger, J. R. Wilson, An improved batch means procedure for simulation output analysis, Management Science 48 (12) (2002) 1569–1586.

[11] E. K. Lada, J. R. Wilson, N. M. Steiger, A wavelet-based spectral method for steady-state simulation analysis, in: Proceedings 2003 Winter Simulation Conference, WSC'03, Washington, 2003.

[12] D. Heidelberger, P. A. Welch, A spectral method for confidence interval generation and run length control in simulations, Communications of the ACM 25 (1981) 175–181.

[13] G. Ewing, D. McNickle, K. Pawlikowski, Spectral analysis for confidence interval estimation under multiple replications in parallel, in: Proceedings 14th European Simulation Symposium, ESS'2002, ISCS Press, Dresden, Germany, 2002, pp. 56–61.

[14] G. C. E. D. McNickle, K. Pawlikowski, Refining spectral analysis for confidence interval estimation in sequential simulation, in: Proceedings 16th European Simulation Symposium, ESS'04, SCSC, Budapest, Hungary, 2004, pp. 75–81.

[15] K. Pawlikowski, D. McNickle, G. Ewing, Coverage of confidence intervals in sequential steady-state simulation, Simulation Practice and Theory 6 (2) (1998) 255–267.

[16] W. D. Kelton, R. P. Sadowski, D. A. Sadowski, Simulation with Arena, McGraw-Hill, N.Y., 2002.

[17] G. Ewing, K. Pawlikowski, D. McNickle, Akaroa 2: Exploiting network computing by distributed stochastic simulation, in: Proceedings 13th European Simulation Multiconference, ESMC'99, SCSC, Warsaw, Poland, 1999, pp. 175–181.

[18] A. M. Law, W. D. Kelton, Simulation Modelling and Analysis, McGraw-Hill, N.Y., 1991.

[19] K. Pawlikowski, V. Yau, D. McNickle, Distributed stochastic discrete-event simulation in parallel time streams, in: Proceedings 1994 Winter Simulation Conference, WSC'94, IEEE Press, Lake Buena Vista, Florida, 1994, pp. 723–730.

[20] G. Ewing, D. McNickle, K. Pawlikowski, Akaroa 2.7.3 User's Manual, University of Canterbury, Christchurch, New Zealand,

www.cosc.canterbury.ac.nz/research/RG/net_sim/simulation_group.html, 2002.

[21] P. L'Ecuyer, Good parameters and implementations for combined multiple recursive random number generators, Operations Research 47 (1) (1999) 159–164.

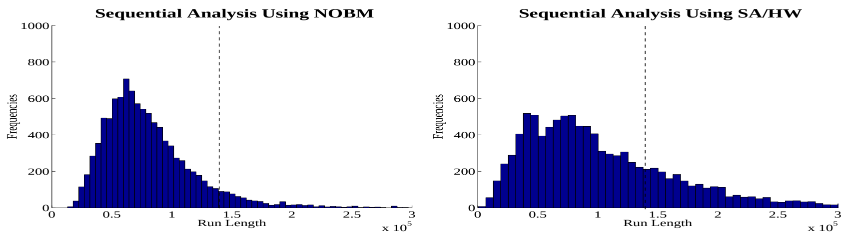
[22] D. J. Daley, The serial correlation coefficients of waiting times in a stationary single server queue, Journal of the Australian Mathematical Society 8 (1968) 683–699.

[23] J. R. Lee, K. Pawlikowski, D. McNickle, Do not trust too short sequential simulation, in: Proceedings 1999 Summer Computer Simulation Conference, SCSM'99, SCSC, Chicago, Illinois, 1999, pp. 97–102.

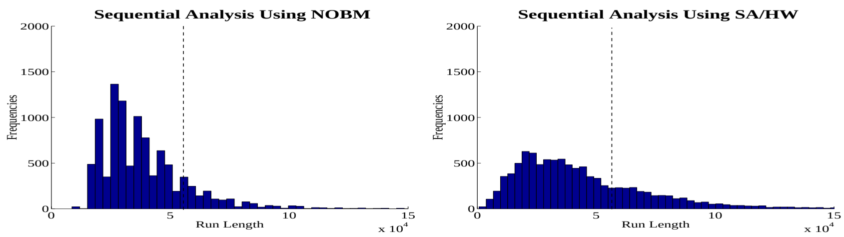
[24] J. R. Lee, K. Pawlikowski, D. McNickle, Sequential steady-state simulation: Rules of thumb for improving accuracy of the final results, in: Proceedings 11th European Simulation Symposium, ESS'99, SCSC, Erlangen, Germany, 1999, pp. 618–622.

[25] D. E. Knuth, Art Of Computer Programming, Vol. 2, Addison-Wesley, N.Y., 1969.

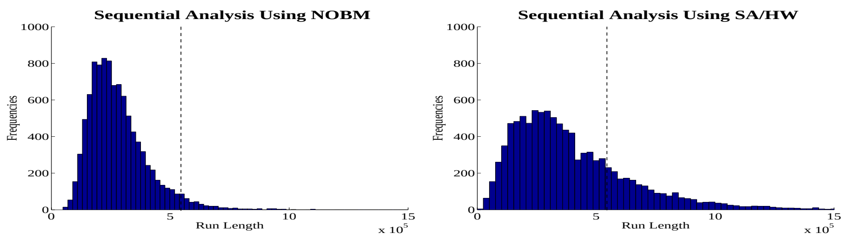
[26] D. Heidelberger, P. A. Welch, Simulation run length control in the presence of an initial transient, Operations Research 31 (1983) 1109–1144.



(a) $M/M/1/\infty$ queueing system



(b) $M/D/1/\infty$ queueing system



(c) $M/H_2/1/\infty$ queueing system

Figure 2: Histograms of simulation run-lengths: each queueing system loaded at 0.9; 10000 replications.

ρ	Number of TSSR	Coverage of TSSR	Probability of being a TSSR	Filtering threshold	Mean run-lengths
			Sequential method of NOBM		
0.1	0	N/A	0.0%	8471	11823
0.2	0	N/A	0.0%	8567	11888
0.3	0	N/A	0.0%	8424	11967
0.4	0	N/A	0.0%	8451	12221
0.5	0	N/A	0.0%	8356	12538
0.6	0	N/A	0.0%	8175	13242
0.7	0	N/A	0.0%	8893	15586
0.8	593	50.6%	5.9%	13318	24826
0.9	1017	35.1%	10.2%	41596	80967
			Sequential method of SA/HW		
0.1	0	N/A	0.0%	1345	1725
0.2	1	100.0%	0.01%	1392	2006
0.3	571	86.0%	5.7%	1549	2493
0.4	1749	77.9%	17.5%	1839	3302
0.5	1069	69.5%	10.7%	2374	4665
0.6	1138	64.1%	11.4%	3356	7277
0.7	1101	53.8%	11.0%	5383	12701
0.8	1000	47.9%	10.0%	10461	27809
0.9	928	38.8%	9.3%	34933	106037

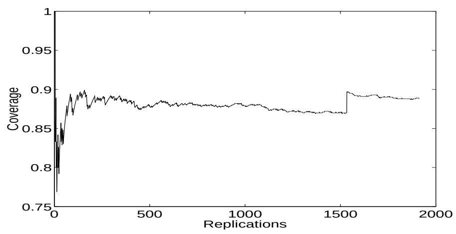
Table 2: $M/M/1/\infty$ queueing system (10,000 replications, estimating the mean response time at a confidence level = 0.95 with maximum statistical error = 10%). TSSR = Too Short Simulation Run.

ρ	Number of TSSR	Coverage of TSSR	Probability of being a TSSR	Filtering threshold	Mean run-lengths
			Sequential method of NOBM		
0.1	0	N/A	0.0%	8526	12043
0.2	0	N/A	0.0%	8520	11967
0.3	0	N/A	0.0%	8473	11986
0.4	0	N/A	0.0%	8438	12099
0.5	0	N/A	0.0%	8389	12313
0.6	0	N/A	0.0%	8307	12685
0.7	0	N/A	0.0%	8466	13517
0.8	0	N/A	0.0%	9470	17024
0.9	532	42.1%	5.3%	19070	39129
			Sequential method of SA/HW		
0.1	1300	94.6%	12.7%	1923	2199
0.2	1110	93.9%	11.1%	1653	1811
0.3	888	92.8%	8.9%	1575	1708
0.4	222	91.4%	2.2%	1512	1701
0.5	0	N/A	0.0%	1423	1851
0.6	39	89.7%	0.4%	1483	2458
0.7	1076	63.3%	10.8%	1975	4218
0.8	928	46.1%	9.3%	3920	10225
0.9	873	34.2%	8.7%	14557	44845

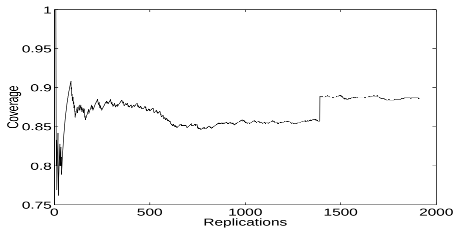
Table 3: $M/D/1/\infty$ queueing system (10000 replications, estimating the mean response time at a confidence level = 0.95 with maximum statistical error = 10%). TSSR = Too Short Simulation Run.

ρ	Number of TSSR	Coverage of TSSR	Probability of being a TSSR	Filtering threshold	Mean run-lengths
			Sequential method of NOBM		
0.1	0	N/A	0.0%	8371	12308
0.2	0	N/A	0.0%	8703	13007
0.3	1240	79.6%	12.4%	10607	15131
0.4	1171	73.3%	11.7%	13595	19200
0.5	1084	67.3%	10.8%	17943	25334
0.6	1346	64.2%	13.5%	24530	34924
0.7	1212	59.8%	12.1%	35658	52293
0.8	1207	52.8%	12.1%	59399	93866
0.9	1084	38.6%	10.8%	153571	281427
			Sequential method of SA/HW		
0.1	1306	69.4%	13.1%	3546	6863
0.2	1228	67.7%	12.3%	5432	10955
0.3	1160	66.2%	11.6%	7586	15768
0.4	1201	64.3%	12.0%	10180	21750
0.5	1151	61.4%	11.5%	13785	30438
0.6	1176	61.3%	11.8%	18865	42893
0.7	1109	56.3%	11.1%	27939	67654
0.8	1141	52.9%	11.4%	49613	126815
0.9	972	43.7%	9.7%	136367	403492

Table 4: $M/H_2/1/\infty$ queueing system (10000 replications, estimating the mean response time at a confidence level = 0.95 with maximum statistical error = 10%). TSSR = Too Short Simulation Run.



(a) Sequential method of NOBM

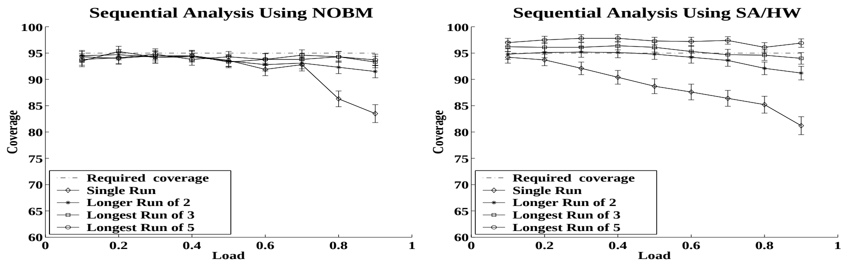


(b) Sequential method of SA/HW

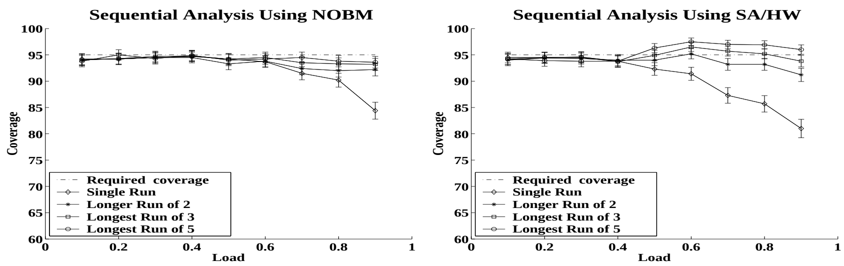
Figure 3: Convergence of coverage for sequential steady-state simulations ($M/M/1/\infty$, load = 0.8)

	Number of runs R	**NOBM** ($\rho = 0.9$)	**SA/HW** ($\rho = 0.4$)
(a) $M/M/1/\infty$	1	$0.102^1 = 0.102$	$0.175^1 = 0.175$
	2	$0.102^2 = 0.0104$	$0.175^2 = 0.0306$
	3	$0.102^3 = 0.0011$	$0.175^3 = 0.0054$
	5	$0.102^5 = 0.00001$	$0.175^5 = 0.00016$
(b) $M/D/1/\infty$	1	$0.053^1 = 0.053$	$0.130^1 = 0.130$
	2	$0.053^2 = 0.0028$	$0.130^2 = 0.0169$
	3	$0.053^3 = 0.0001$	$0.130^3 = 0.0022$
	5	$0.053^5 =$ 4e-7	$0.130^5 = 0.00004$
(c) $M/H_2/1/\infty$	1	$0.135^1 = 0.135$	$0.131^1 = 0.131$
	2	$0.135^2 = 0.0182$	$0.131^2 = 0.0172$
	3	$0.135^3 = 0.0025$	$0.131^3 = 0.0022$
	5	$0.135^5 = 0.00004$	$0.131^5 = 0.00004$

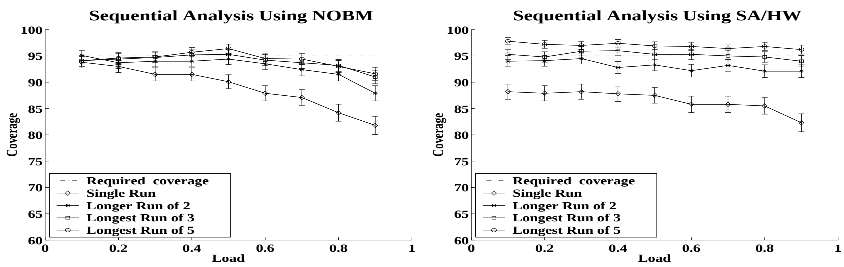
Table 5: The worst case of the probability of R independent replications belonging to the class of 'too short' simulation runs (theoretical confidence level = 0.95)



(a) $M/M/1/\infty$ queueing system

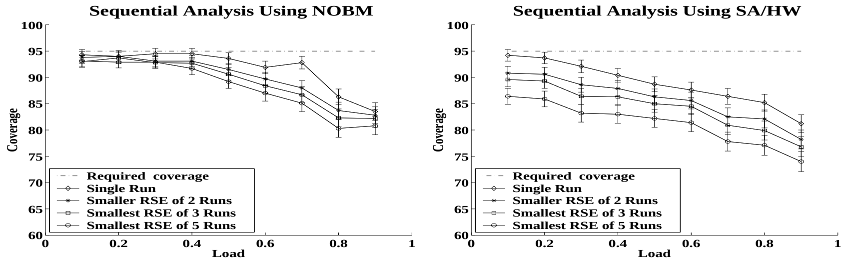


(b) $M/D/1/\infty$ queueing system

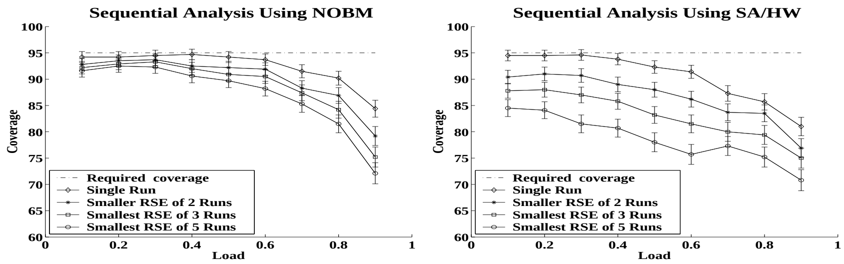


(c) $M/H_2/1/\infty$ queueing system

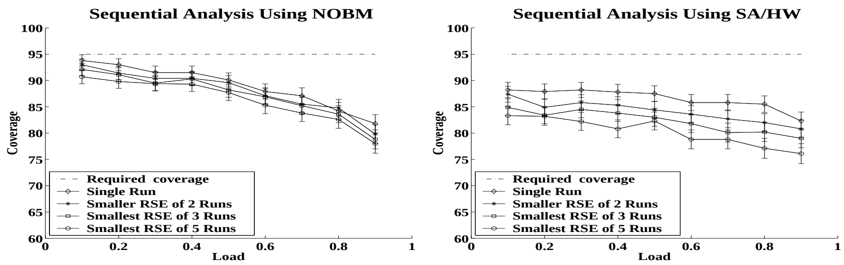
Figure 4: Coverage of the CIs with Rule I (take the longest of R replications; R = 1, 2, 3 and 5)



(a) $M/M/1/\infty$ queueing system

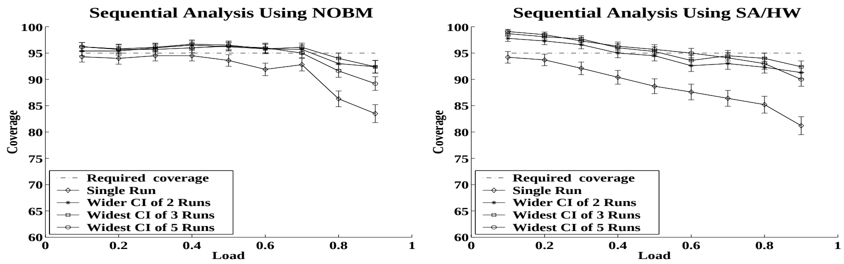


(b) $M/D/1/\infty$ queueing system

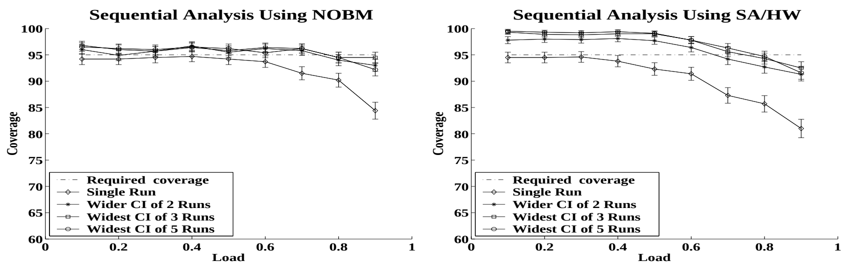


(c) $M/H_2/1/\infty$ queueing system

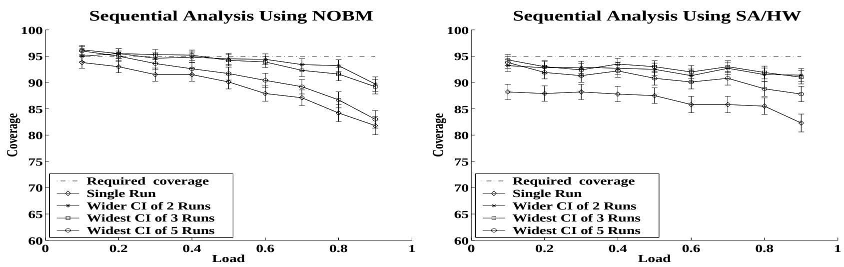
Figure 5: Coverage of the CIs with Rule II (take the most accurate result out of R results obtained; R = 1, 2, 3 and 5)



(a) $M/M/1/\infty$ queueing system

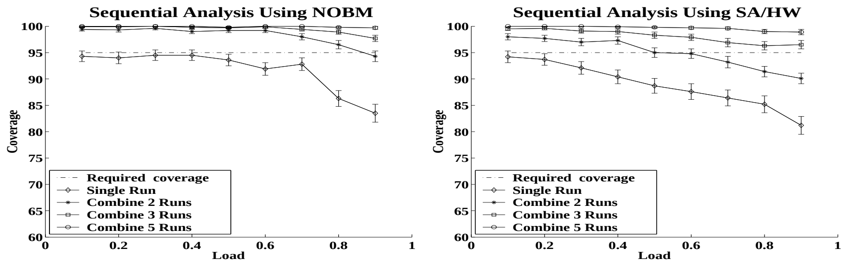


(b) $M/D/1/\infty$ queueing system

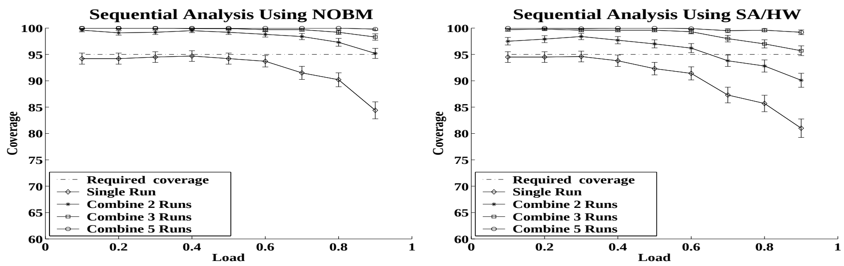


(c) $M/H_2/1/\infty$ queueing system

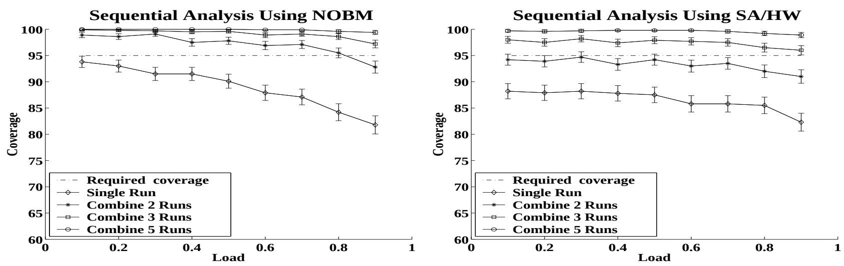
Figure 6: Coverage of the CIs with Rule III (take the widest CIs of R replications; $R =$ 1, 2, 3 and 5)



(a) $M/M/1/\infty$ queueing system

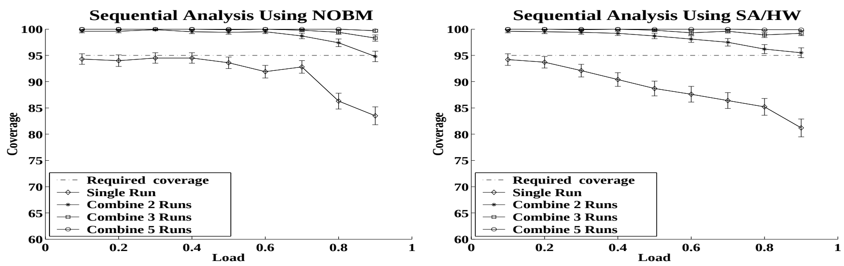


(b) $M/D/1/\infty$ queueing system

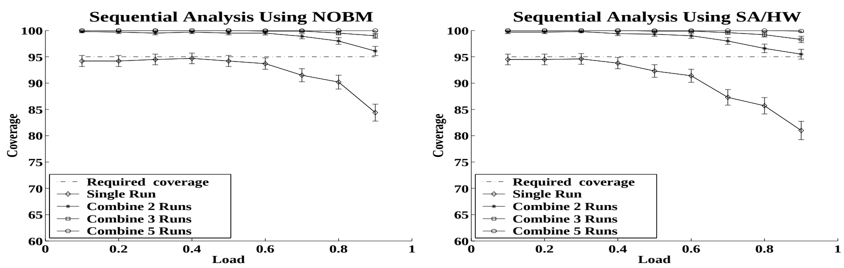


(c) $M/H_2/1/\infty$ queueing system

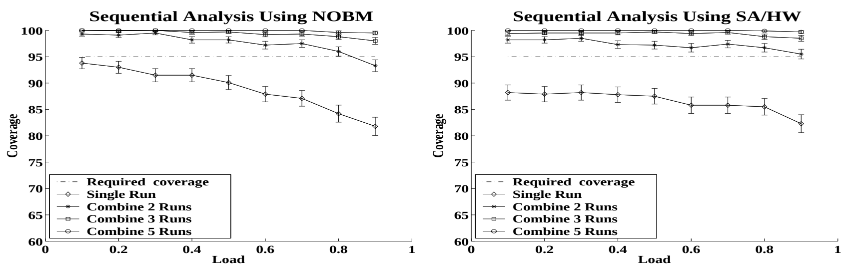
Figure 7: Coverage of the CIs with Rule IV (combining R replications; $R = 1$, 2, 3 and 5)



(a) $M/M/1/\infty$ queueing system



(b) $M/D/1/\infty$ queueing system



(c) $M/H_2/1/\infty$ queueing system

Figure 8: Coverage of the CIs with Rule V (combination of Rules III and IV)