Proceedings of the 2006 Winter Simulation Conference
L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, eds.

A NEW APPROACH FOR PARALLEL STEADY-STATE SIMULATIONS

Ming-hua Hsieh

Department of Management Information Systems
National Chengchi University
Taipei, 11623, TAIWAN

ABSTRACT

We propose a new procedure for building confidence interval estimators of steady-state parameters in discrete event simulations. The procedure uses parallel processors to generate independent replications and constructs the confidence interval estimator by solving a generalized least square problem. The most appealing theoretical feature of the proposed procedure is that the precision of the resulted estimator can be improved by simply increasing the number of processors (or independent replications) while the simulated time length is fixed on an appropriate level on each processor. Experiments conducted on M/M/1 queue waiting time processes in heavy traffic confirm this theoretical property.

1 INTRODUCTION

Let $Y = (Y(t) : t \geq 0)$ be a real-valued stochastic process representing the output of a discrete event simulation. Suppose that Y satisfies a law of large number (LLN) of the form

$$\alpha(t) \equiv \frac{1}{t}\int_0^t Y(s)ds \Rightarrow \alpha \tag{1}$$

as $t \to \infty$, for some constant α, where $\Rightarrow$ denotes convergence in distribution. The steady-state simulation problem is concerned with the efficient estimation of the steady-state mean of α, and the construction of associated confidence intervals (CIs).

As suggested by (1), the time-average $\alpha(t)$ is an obvious point estimator of α. Suppose that p processors are available for simulation, and each processor simulates the process Y independently up to (deterministic) time t. Let $\alpha_i(t)$ denote the time-average $\alpha(t)$ generated by processor i. Set

$$\alpha(t,p) = \frac{1}{p}\sum_{i=1}^{p}\alpha_i(t), \tag{2}$$

and

$$S^2(t,p) = \frac{1}{p-1}\sum_{i=1}^{p}(\alpha_i(t) - \alpha(t,p))^2. \tag{3}$$

Then one might expect

$$\alpha(t,p) \pm t_{1-\gamma/2,p-1}S(t,p) \tag{4}$$

is an asymptotically valid $100(1-\gamma)\%$ CI for α, and the absolute precision of the estimator (usually defined by the half-length of the CI) is proportional to the order of $\sqrt{p}$. We call estimator (4) the standard estimator. The standard estimator will converge to a wrong value if simulated time t and the number of processors p are not chosen appropriately (Glynn and Heidelberger 1991b, Glynn and Heidelberger 1992a, and Glynn and Heidelberger 1992b.) Such statistical problems basically arise because any bias effects on a single replication are magnified on multiple replications. This type of problems also arise in transient simulation context; see (Heidelberger 1988) and (Glynn and Heidelberger 1991a), for example.

The process Y is typically initialized via a distribution for $Y(0)$ that is not characteristic of the steady-state behavior. As a consequence, $\alpha(t)$ is biased as an estimator of α. In other words, $E\alpha(t) \neq \alpha$. For the same reason, $E\alpha(t,p) \neq \alpha$.

The bias in $\alpha(t)$ as an estimator of α is known as the "initial bias". The initial bias problem, in the single processor context, can be mitigated in two different ways. One approach is to delete that initial segment of the simulation that is "contaminated" by initial bias. Such an initial bias deletion approach has been studied by many authors; see, for example, Cash et al. (1992), Glynn (1995), Goldsman, Schruben, and Swain (1994), Schruben (1982), Schruben et al. (1983), White (1997), and White et al. (2000). An alternative is to consider an estimator, based on simulating Y over $[0,t]$, that attempts to compensate for the bias present in $\alpha(t)$. We refer to such estimators as "bias reducing" estimators. The bias reducing estimators usually need to

make use of independent identically distributed quantities. Exploiting the regenerative structure of the process Y is a commonly used approach; see Hsieh et al. (2004) for a survey.

In the parallel processors (multiple replications) context, Glynn and Heidelberger (1992a) and Glynn and Heidelberger (1992b) have studied the initial bias deletion approach. Both theoretical and empirical results in their study show that the standard estimator (4) and its variant with initial bias deletion are not statistic efficient and ratio estimators are more appropriate.

The proposed estimator in this paper is a bias reducing estimator, thus requires no initial bias deletion. In addition, multiple replications provide independent identically distributed quantities. Therefore, the proposed estimator does not need to explore the regenerative structure of the process Y. The most appealing theoretical property of the proposed estimator is that the precision of the estimator can be improved by simply increasing p while the simulated time t is fixed at an adequate level.

This paper is organized as follows. In Section 2, we describe the proposed estimator and discuss its theoretical properties. In Section 3, we discuss some of our computational experience with the procedures introduced in Section 2. Finally, Section 4 offers some concluding remarks.

2 THE PROPOSED ESTIMATOR

Suppose that Y is the simulation output that is derived from the simulation of a stochastic system that can be modeled in terms of a Markov process X. In particular, suppose that $Y(t) = f(X(t))$, where $X = (X(t) : t \geq 0)$ is a Markov process living on a state space S, and $f : S \to \Re$ is a real-valued performance measure. Assuming that X exhibits positive recurrent behavior, it can be shown in substantial generality that

$$E\alpha(t) = \alpha + \frac{b_1}{t} + o(e^{-\beta t}) \tag{5}$$

as $t \to \infty$, for some constants b and β (where $\beta > 0$), where $o(a(t))$ denotes a function $f(t)$ such that $f(t)/a(t) \to 0$ as $t \to \infty$. See, for example, Glynn (1984) for such a result in the setting of finite-state continuous-time Markov chains.

In this paper, we impose a more mild requirement on the bias function. To be precise, let $b(t) = E\alpha(t) - \alpha$. The requirement of $b(\cdot)$ is that

$$b(t) \to 0, \quad \text{as } t \to \infty. \tag{6}$$

Set $h = 1/t$. The above equation is equivalent to

$$b(h) \to 0, \quad \text{as } h \to 0.$$

Thus, a simple application of Taylor expansion gives

$$b(t) = \frac{b_1}{t} + o(1/t), \quad \text{or} \tag{7}$$

$$b(t) = \frac{b_1}{t} + \frac{b_2}{t^2} + o(1/t^2), \tag{8}$$

where b_1 and b_2 are the coefficients of the first order and second order terms in the Taylor Expansion.

Equations (7) and (8) suggest that $b(t)$ can be approximated by b_1/t or $b_1/t + b_2/t^2$. Since we do not expect that the simulated time t on each processor is long, Approximation (8) seems a better choice for approximating $b(t)$. Therefore, in the subsequent analysis, we will use $b_1/t + b_2/t^2$ to approximate $b(t)$. Note that this approximation does not require the simulated time t be very large. As long as the absolute difference between $b(t)$ and $b_1/t + b_2/t^2$ is much smaller than the required precision of the CI, this approximation has little impact on the quality of the CI.

Assume the simulated time on each of the p processors is T and the time-average $\alpha(t)$ up to time t ($t \leq T$) on processor i ($1 \leq i \leq p$) is denoted by $\alpha_i(t)$. Let $\sigma^2(t)$ be the variance of $\alpha(t)$ and set

$$\alpha(t,p) = \frac{1}{p}\sum_{i=1}^{p} \alpha_i(t), \tag{9}$$

for $0 < t \leq T$.

Since $\alpha_i(t)$'s are independent, it is easy to see that $\alpha(t,p)$ satisfies a central limit theorem (CLT)

$$\frac{\sqrt{p}(\alpha(t,p) - \alpha - b(t))}{\sigma(t)} \Rightarrow N(0,1), \tag{10}$$

as $p \to \infty$, where $N(0,1)$ denotes the standard normal random variable.

Suppose that $0 < t_1 < t_2 < \cdots < t_n = T$ and $|b(t) - (b_1/t + b_2/t^2)|$ is negligible for $t \geq t_1$. Let

$$\tilde{\alpha}(p) = (\alpha(t_1,p), \alpha(t_2,p), \ldots, \alpha(t_n,p))^T,$$

$$A = \begin{pmatrix} 1 & 1/t_1 & 1/t_1^2 \\ 1 & 1/t_2 & 1/t_2^2 \\ \vdots & \vdots & \vdots \\ 1 & 1/t_n & 1/t_n^2 \end{pmatrix}$$

and

$$b = (\alpha, b_1, b_2)^T.$$

Then, it is straightforward to show that

$$E\tilde{\alpha}(p) = Ab.$$

Let S be the covariance matrix of the random column vector $(\alpha(t_1), \alpha(t_2), \ldots, \alpha(t_n))^T$, then the covariance matrix of $\tilde{\alpha}(p)$ is S/p. Note that $\tilde{\alpha}(p)$ is an approximate multivariate normal random variable by the CLT (10). Thus, we have the following linear model

$$\tilde{\alpha}(p) = Ab + \varepsilon, \tag{11}$$

where ε is normally distributed error vector with mean 0 and covariance matrix S/p. The linear model (12) is the vehicle for building the confidence interval for α. Before we proceed to the procedure of constructing he confidence interval for α, we need the following proposition.

Proposition 1 *Given a full rank univariate linear model*

$$\theta = Ab + \varepsilon \tag{12}$$

for the $n \times 1$ random vector a, where A is an $n \times k$ matrix of rank $k \leq n$, b is an $k \times 1$ (unknown) parameter vector, and ε is the $n \times 1$ normally distributed error vector with mean 0. If the covariance matrix of ε is Σ and consider θ a single observation from the linear model (12), then:

1. *The maximum likelihood estimator of b is*

$$\hat{b} = (A^T \Sigma^{-1} A)^{-1} A^T \Sigma^{-1} \theta. \tag{13}$$

   Estimator $\hat{b}$ is also known as generalized least square estimator.
2. *Estimator $\hat{b}$ is an unbiased estimator for b, i.e., $E\hat{b} = b$.*
3. *The covariance matrix of the generalized least square estimator $\hat{b}$ is*

$$\Sigma_{\hat{b}\hat{b}} = (A^T \Sigma^{-1} A)^{-1}.$$

4. *For $1 \leq i \leq n$,*

$$\frac{\hat{b}(i) - b(i)}{\sqrt{\Sigma_{\hat{b}\hat{b}}(i,i)}} \sim t_{n-k},$$

   where t_n denotes t-distributed random variable with n degrees of freedom.

We are now ready to describe the proposed procedure:

1. Input p, T, $0 < t_1 < t_2 < \cdots < t_n = T$, and the required confidence level $1 - \gamma$.
2. Generate independent replicate of Y up to time T on each processor and collect $\alpha_i(t_1), \alpha_i(t_2), \ldots, \alpha_i(t_n)$ on each processor i, $1 \leq i \leq p$.
3. Compute the sample covariance matrix $\hat{S}$ of $(\alpha(t_1), \alpha(t_2), \ldots, \alpha(t_n))^T$ by the data collected in step 2.
4. Compute $\hat{b}$ and $\Sigma_{\hat{b}\hat{b}}$ according to Proposition 1.
5. Output the confidence interval of α by

$$\hat{b}(1) \pm t_{1-\gamma/2, n-3} \sqrt{\Sigma_{\hat{b}\hat{b}}(1,1)}. \tag{14}$$

The most appealing theoretical feature of the procedure above is that the precision of the resulted estimator can be improved by simply increasing p while the simulated time T is fixed. This theoretical feature can be summarized in the following theorem.

Theorem 1 *Suppose that $0 < t_1 < t_2 < \cdots < t_n = T$ and $|b(t) - (b_1/t + b_2/t^2)|$ is negligible for $t \geq t_1$. Assume $\Sigma_{\hat{b}\hat{b}}(p)$ denotes the covariance matrix of $\hat{b}$ when the number of processors is p and m a positive integer. Then*

$$\Sigma_{\hat{b}\hat{b}}(mp) = \frac{1}{m} \Sigma_{\hat{b}\hat{b}}(p).$$

Let $hl(p)$ denote the half length of CI when the number of processors is p. By (14), above equation implies

$$hl(mp) = \frac{1}{\sqrt{m}} hl(p).$$

Proof Since

$$\Sigma_{\hat{b}\hat{b}}(mp) = (A^T (S/mp)^{-1} A)^{-1} = \frac{1}{m} (A^T (S/p)^{-1} A)^{-1},$$

we have

$$\Sigma_{\hat{b}\hat{b}}(mp) = \frac{1}{m} \Sigma_{\hat{b}\hat{b}}(p).$$

□

3 EMPIRICAL RESULTS

We chose the waiting time process in the $M/M/1$ queue with server utilization $\rho = 0.9$ and an empty-and-idle initial condition as the test problem. This is a particularly difficult test problem. Steiger et al. (2005) state several reasons:

1. the initialization bias is large and decays relatively slowly;
2. in steady-state operation the autocorrelation function of the waiting time process decays very slowly with increasing lags; and
3. in steady-state operation the marginal distribution of waiting times has an exponential tail and is therefore markedly nonnormal.

Thus we expect the proposed estimator will perform well on most of real world applications if it does well on this test problem.

The crucial factors of the proposed estimator include the number of processors p, the simulated time T on each processor and the time points $t_1,\ldots,t_n$ to formulate the linear model. We selected 18 test cases by varying the value of each factor. In particular, they include the following combinations:

1. $p = 512, 2048$, and 8192.
2. $T = 1000$ and 2000.
3. Three different selections of the time points $t_1,\ldots,t_n$. The time points were selected by first fixing n and t_1 and then the time points are equally spaced within the interval (t_1, T). We first selected $n = 6, 12$, or 18; and then chose t_1 such that t_1 falls in the interval $[200, 300]$.

Table 1 and Table 2 show the detail information of these test cases.

We performed 1000 independent replications of nominal 90% and 95% confidence intervals for each test case. From these independent replications of confidence intervals, we compute:

1. the empirical coverage probability; since the nominal confidence level are 90% and 95%, the corresponding standard errors of the empirical coverage probabilities are $\sqrt{0.9 * 0.1/1000} \approx 0.95\%$ and $\sqrt{0.95 * 0.05/1000} = 0.69\%$ respectively;
2. the average (avg.) of the half-length of these confidence intervals;
3. the standard deviation (s.d.) of the half-length of these confidence intervals.

The empirical results are shown in Table 1 and 2. Below we discuss the observed properties of the proposed estimator.

- The confidence interval coverage is reasonably accurate for most of the test cases. To be precise, let p be the true coverage probability of the proposed confidence interval estimator and we consider testing the following hypotheses:

$$\begin{aligned} H_0 : p &= \text{nominal value}, \\ H_1 : p &\neq \text{nominal value}. \end{aligned}$$

  Then under 1% significance level, only 3 out 36 tests (see Table 1 and 2) can reject H_0. This suggests the proposed estimator is not very sensitive to the choice of T and $(t_1,\ldots,t_n)$.
- The number of time points n has effect on the avg. and the s.d. of the half-length of these confidence intervals. Both avg. and s.d. of the half-length of CIs decrease when n increases. We expect that both of them will converge to limits when n goes to ∞. However, from an applied standpoint, we should not choose large n, because the cost of computing the solution of the corresponding linear model will be too high.
- The effect of p on the width of the CIs follows Theorem 1 exactly. For example, in Table 1, let us consider the avg. CI half lengths for $(t_1,\ldots,t_n) = (250, 400, \ldots, T)$ and nominal 90% confidence interval. The avg. half-lengths of CIs are 0.718 (for $p = 512$), 0.357 (for $p = 2048$), and 0.185 (for $p = 8196$). Both $0.718/0.357 \approx 2.01$ and $0.357/0.185 \approx 1.93$ are close to the theoretical value 2. This property is very desirable, because it suggests a p-fold speedup is achievable.

4 CONCLUSIONS

We propose a new procedure for building confidence interval estimators of steady-state parameters in discrete event simulations. The procedure uses parallel processors to generate independent replications and constructs the confidence interval estimator by solving a generalized least square problem. The most appealing theoretical feature of the proposed procedure is that the precision of the resulted estimator can be improved by simply increasing the number of processors while the simulated time length is fixed at an proper level on each processor. The experiments we conducted on M/M/1 queue waiting time processes in heavy traffic also support this theoretical property.

In addition to this main advantage, the proposed estimator also has some other advantages:

1. The procedure is simple to implement. Compared to the standard estimator (4), extra computational works includes only the computation of the sample covariance matrix $\hat{S}$ and solving a (small-size) generalized least square problem. The procedure can also be easily adopted by a distributed computing environment.
2. The total computational effort is comparable to the latest single processor estimators. For example, ASAP3 (Steiger et al. 2005) requires an average sample size of 969,011 to achieve a average CI half-length of 0.32 for the test problem; whereas, the proposed procedure requires a sample size of 1,024,000 ($p = 512$, $T = 2000$, and $n = 18$) to achieve a average CI half-length of 0.36 for the test problem.

Table 1: Empirical Performance of the Proposed Procedure for the $M/M/1$ Queue Waiting Time Process with $\rho = 0.9$ and Simulated Time $T = 1000$ Based on 1000 Independent Replications of Nominal 90% and 95% Confidence Intervals

$p = 512$; $n = 6, 12,$ or 18	Nominal 90% CIs	Nominal 95% CIs
$(t_1, \ldots, t_n) = (250, 400, \ldots, T)$		
coverage	89.1%	94.3%
avg. CI half-length	0.718	0.972
s.d. CI half-length	0.317	0.428
$(t_1, \ldots, t_n) = (230, 300, \ldots, T)$		
coverage	90.2%	95.0%
avg. CI half-length	0.528	0.652
s.d. CI half-length	0.134	0.165
$(t_1, \ldots, t_n) = (235, 280, \ldots, T)$		
coverage	85.7%	92.0%
avg. CI half-length	0.507	0.616
s.d. CI half-length	0.103	0.125
$p = 2048$; $n = 6, 12,$ or 18		
$(t_1, \ldots, t_n) = (250, 400, \ldots, T)$		
coverage	89.4%	93.5%
avg. CI half-length	0.357	0.483
s.d. CI half-length	0.155	0.209
$(t_1, \ldots, t_n) = (230, 300, \ldots, T)$		
coverage	91.2%	96.4%
avg. CI half-length	0.269	0.331
s.d. CI half-length	0.066	0.082
$(t_1, \ldots, t_n) = (235, 280, \ldots, T)$		
coverage	88.1%	93.6%
avg. CI half-length	0.256	0.311
s.d. CI half-length	0.049	0.059
$p = 8192$; $n = 6, 12,$ or 18		
$(t_1, \ldots, t_n) = (250, 400, \ldots, T)$		
coverage	90.6%	95.6%
avg. CI half-length	0.185	0.250
s.d. CI half-length	0.078	0.106
$(t_1, \ldots, t_n) = (230, 300, \ldots, T)$		
coverage	89.9%	94.4%
avg. CI half-length	0.140	0.173
s.d. CI half-length	0.034	0.042
$(t_1, \ldots, t_n) = (235, 280, \ldots, T)$		
coverage	89.3%	94.5%
avg. CI half-length	0.132	0.161
s.d. CI half-length	0.025	0.030

Table 2: Empirical Performance of the Proposed Procedure for the $M/M/1$ Queue Waiting Time Process with $\rho = 0.9$ and Simulated Time $T = 2000$ Based on 1000 Independent Replications of Nominal 90% and 95% Confidence Intervals

$p = 512$; $n = 6, 12,$ or 18	Nominal 90% CIs	Nominal 95% CIs
$(t_1, \ldots, t_n) = (250, 600, \ldots, T)$		
coverage	89.4%	95.1%
avg. CI half-length	0.489	0.661
s.d. CI half-length	0.201	0.272
$(t_1, \ldots, t_n) = (350, 500, \ldots, T)$		
coverage	88.4%	94.0%
avg. CI half-length	0.404	0.499
s.d. CI half-length	0.100	0.124
$(t_1, \ldots, t_n) = (300, 400, \ldots, T)$		
coverage	88.0%	94.3%
avg. CI half-length	0.362	0.439
s.d. CI half-length	0.073	0.089
$p = 2048$; $n = 6, 12,$ or 18		
$(t_1, \ldots, t_n) = (250, 600, \ldots, T)$		
coverage	89.1%	94.7%
avg. CI half-length	0.244	0.329
s.d. CI half-length	0.102	0.137
$(t_1, \ldots, t_n) = (350, 500, \ldots, T)$		
coverage	90.5%	94.7%
avg. CI half-length	0.207	0.255
s.d. CI half-length	0.050	0.062
$(t_1, \ldots, t_n) = (300, 400, \ldots, T)$		
coverage	90.8%	95.7%
avg. CI half-length	0.187	0.227
s.d. CI half-length	0.035	0.042
$p = 8192$; $n = 6, 12,$ or 18		
$(t_1, \ldots, t_n) = (250, 600, \ldots, T)$		
coverage	88.5%	94.0%
avg. CI half-length	0.125	0.168
s.d. CI half-length	0.054	0.072
$(t_1, \ldots, t_n) = (350, 500, \ldots, T)$		
coverage	86.8%	93.4%
avg. CI half-length	0.103	0.127
s.d. CI half-length	0.024	0.030
$(t_1, \ldots, t_n) = (300, 400, \ldots, T)$		
coverage	89.8%	93.7%
avg. CI half-length	0.092	0.112
s.d. CI half-length	0.018	0.021

REFERENCES

Cash, C. R., B. L. Nelson, D. G. Dippold, J. M. Long, and W. P. Pollard. 1992. Evaluation of tests for initial-condition bias. In *Proceedings of the 24th Winter Simulation Conference (WSC'92)*, 577–585.

Glynn, P. W. 1984. Some asymptotic formulas for markov chain with applications to simulation. *Journal of Statistical Computation and Simulation* 19:97–112.

Glynn, P. W. 1995. Some new results on the initial transient problem. In *Proceedings of the 27th Winter Simulation Conference (WSC'95)*, 165–170.

Glynn, P. W., and P. Heidelberger. 1991a. Analysis of initial transient deletion for replicated steady-state simulations. *Operations Research Letters* 10:437–443.

Glynn, P. W., and P. Heidelberger. 1991b. Analysis of parallel, replicated simulations under a completion time constraint. *ACM Transations on Modeling and Computer Simulation* 1:3–23.

Glynn, P. W., and P. Heidelberger. 1992a. Analysis of initial transient deletion for parallel steady-state simulations. *SIAM Journal on Scientific and Statistical Computing* 13:909–922.

Glynn, P. W., and P. Heidelberger. 1992b. Experiments with initial transient deletion for parallel, replicated steady-state simulations. *Management Science* 38:400–418.

Goldsman, D., L. W. Schruben, and J. Swain. 1994. Tests for transient means in simulated time series. *Naval Research Logistics Quarterly* 41:171–187.

Heidelberger, P. 1988. Discrete event simulations and parallel processing: statistical properties. *SIAM Journal on Scientific and Statistical Computing* 9:1114–1132.

Hsieh, M.-H., D. L. Iglehart, and P. W. Glynn. 2004. Empirical performance of bias-reducing estimators for regenerative steady-state simulations. *ACM Transations on Modeling and Computer Simulation* 14:325–343.

Schruben, L. W. 1982. Detecting initialization bias in simulation output. *Opertions Research* 30 (3): 151–153.

Schruben, L. W., H. Singh, and L. Tierney. 1983. Optimal tests for initialization bais in simulation output. *Opertions Research* 31 (6): 1167–1178.

Steiger, N. M., E. K. Lada, J. R. Wilson, J. A. Joines, C. Alexopoulos, and D. Goldsman. 2005. ASAP3: a batch means procedure for steady-state simulation analysis. *ACM Transations on Modeling and Computer Simulation* 15 (1): 39–73.

White, K. P. 1997. An effective truncation heuristic for bias reduction in simulation output. *Simulation* 69 (6): 323–334.

White, K. P., M. J. Cobb, and S. C. Spratt. 2000. A comparison of five steady-state truncation heuristics for simulation. In *Proceedings of the 32nd Winter Simulation Conference (WSC'00)*, 755–760.

AUTHOR BIOGRAPHY

MING-HUA HSIEH is an Associate Professor of the Department of Management Information Systems of National Chengchi University, Taiwan. From 1997 to 1999, he was a software designer at Hewlett Packard company, California. He is a member of INFORMS. His research interests include simulation methodology and financial engineering. His e-mail address is <mhsieh@mis.nccu.edu.tw>.