ON DATABASED CHOICE OF BATCH SIZE FOR SIMULATION OUTPUT ANALYSIS

Michael Sherman

Department of Statistics
Texas A&M University
College Station, TX, 77843, U.S.A.

ABSTRACT

The use of batch means is a well known technique for estimating the variance of point estimators computed from simulation experiments. The batch means variance estimator is simply the (appropriately scaled) sample variance of the estimator computed on subsets of consecutive observations. For the method to be practical, a good choice of batch length is necessary. We propose a method to estimate the optimal batch length using only the observed data. In contrast to results which model the unknown underlying dependence structure in terms of a few unknown parameters (e.g., autoregression), the method is completely model free. The proposed algorithm makes use of previous asymptotic results giving the order of batch length as a function of the simulation length in order to calibrate an empirical estimate of batch length for a shorter simulation length. We describe the algorithm, present three numerical studies which demonstatrates its efficacy, and discuss large sample consistency.

1 INTRODUCTION

Consider the following scenario: We observe the output sequence $\{X_i : 1 \leq i \leq n\}$ from a simulation run, and we desire an estimate of μ, the mean of the system. A natural estimator is the unbiased observed mean from the simulation:

$$\overline{X}_n := \sum_{i=1}^{n} X_i/n.$$

Having computed this, a second and more formidable goal is to obtain a range of plausible values for μ, i.e., a confidence interval. Typically, the standardized mean converges to the normal distribution, i.e.:

$$n^{1/2}(\overline{X}_n - \mu) \xrightarrow{D} N(0, V),$$

where $\xrightarrow{D}$ denotes convergence in distribution, and thus $(\overline{X}_n - 1.96(V/n)^{1/2}, \overline{X}_n + 1.96(V/n)^{1/2})$ is an approximate 95% confidence interval for μ. The problem is that $V := \sum_{i=-\infty}^{\infty} Cov(X_0, X_i) := \sum_{i=-\infty}^{\infty} \gamma(i)$ is typically unknown and thus needs to be estimated. Alternatively, in place of a confidence interval we may want an estimate of V to simply gauge the stability of $\overline{X}_n$.

Due to the potential serial dependence in the output sequence, estimating V is a nontrivial task. Estimators based on reusing the data, variously known as batch, subsampling, resampling, estimators have been proposed. We will refer to all these methods as batch methods. The basic idea of all batch methods is similar: compute batch means, i.e., the mean of smaller series of consecutive observations as "replicates" of $\overline{X}_n$, and compute a standardized sample variance of these replicates to estimate $Var(\overline{X}_n)$. It should be noted that there are a wide variety of alternatives to batching, e.g., spectral analysis, time series models, and regeneration. These methods are discussed by, e.g., Fishman (1978) and Bratley, Fox, and Schrage (1987). In the sequel we only consider batch variance estimators.

For a variance estimator computed from batch means to be practical, a good choice of batch length is necessary. In Section 2 we introduce various batch estimators and discuss their asymptotic properties. It turns out that a large class of estimators share similar asymptotic properties. In Section 3, we use this observation to form a databased, empirical method for choosing batch length and we study the method in three different settings: moving average output (MA(1)), autoregressive output, (AR(1)), and output from an M/M/1 queue. These examples demonstrate the finite sample efficacy of the proposed method. The Appendix gives some theoretical justification for the proposed algorithm.

2 BATCH MEANS VARIANCE ESTIMATORS

Let $\{X_i : 1 \le i \le n\}$ denote the simulation output and assume that these observations come from a stationary sequence, i.e., the simulation has reached steady state. Let

$$X_b^i := (X_{i+1}, ..., X_{i+b}), i = 0, ..., n-b,$$

denote the "subseries" or "batches" of length b starting with the $(i+1)$'th observation, so that in particular, X_n^0 denotes the entire output from the simulation. The important point is that for each i, X_b^i maintains the same dependence structure as the original sequence X_n^0. Let $\overline{X}_b^i := \sum_{j=i+1}^{i+b} X_i/b, \quad i = 0, ..., n-b,$ denote the corresponding batch means.

Two natural estimators of $nVar(\overline{X}_n)$ are $\hat{V}_N$, based on using all possible nonoverlapping replicates and $\hat{V}_O$, based on all possible overlapping replicates. These are defined as:

$$\hat{V}_N := b \sum_{i=0}^{k-1} \frac{(\overline{X}_b^{ib} - \overline{X}_n)^2}{k},$$

where k is the largest integer less than n/b, and

$$\hat{V}_O := b \sum_{i=0}^{n-b} \frac{(\overline{X}_b^{i} - \overline{X}_n)^2}{(n-b+1)}.$$

It should be noted that various authors use slightly different scaling constants in the definitions of $\hat{V}_N$ and $\hat{V}_O$, but all are reasonably close for large b and small values of the ratio b/n.

If $\hat{V}_N$ is used in the confidence interval described in the introduction a more accurate interval is: $(\overline{X}_n \pm t_{(k-1),.025}(\hat{V}_N/n)^{1/2})$ (by Glynn and Inglehart, 1990) while the interval based on $\hat{V}_O$ should be $(\overline{X}_n \pm t_{(k-1)(3/2),.025}(\hat{V}_N/n)^{1/2}$, where $t_{v,\alpha}$ denotes the $(1-\alpha)100^{th}$ percentile of the t-distribution with v degrees of freedom. Different users suggest using different degrees of freedom (see, e.g., Sargent, Kang, and Goldsman, 1992).

Other estimators like Standardized Time Series (Schruben, 1983) and spaced batch means (Fox, Goldsman, and Swain, 1991) can also be construcuted from batch means. See also, Pedrosa and Schmeiser (1993).

An estimator of $V, \hat{V}$, will be said to be L_2 consistent if $E(\hat{V} - V)^2 \to 0$ as $b \to \infty$ and $b/n \to 0$.

In order for an estimator of V to be consistent the output from the simulation needs to satisfy some type of "weak dependence" condition. One such condition is the strong mixing condition. This assumption allows for correlation between observations at arbitrarily long time lags, yet certain moment conditions on the marginal distribution of X_i guarantee that $V < \infty$. Examples of strongly mixing sequences are moving averages and autoregressive sequences with normal, double exponential, or Cauchy errors by the results of Gastwirth and Rubin (1975). Roughly, the simulation is strong mixing if dependence between observations becomes weak at long lags.

It should be pointed out that the batch means methodology can be extended to estimate the variance of arbitrary statistics. For example, fearing that the simulation has large variability (i.e., the distribution of X_i is heavy tailed) we may choose to estimate μ by the sample median, or more generally by a linear combination of order statistics, i.e., an L-Estimator (see, e.g., Serfling, 1980). In this setting there are fewer alternatives to batch methodology due to analytic intractability, yet the batch method works in a completely analogous manner to the case of the sample mean.

Specifically, let $s_n := s_n(X_1, ..., X_n)$ be any statistic (robust or adaptive), which estimates a real valued parameter, θ. Assuming that the statistic is asymptotically normally distributed, we need an estimate of $Var(s_n)$ in order to obtain a confidence interval for θ.

Let s_b^i denote the corresponding subseries replicates, or batch statistics, $s_b^i := s_b(X_b^i)$. Then the variance estimator can be formed as in the case of $\overline{X}_n$ by simply replacing $\overline{X}_b^i$ by s_b^i. For example, the estimator of $nVar(s_n)$ based on nonoverlapping batch statistics is simply:

$$\hat{V}_N := b \sum_{i=0}^{k-1} \frac{(s_b^{ib} - s_n)^2}{k}.$$

To simplify the exposition it will be helpful to focus on one particular variance estimator. To decide which one we compare the biases and variances of the estimators. For a given batch length, $\hat{V}_N$ and $\hat{V}_O$ have approximately the same bias, so we focus on the variance of the estimators.

2.1 Comparing $\hat{V}_N$ and $\hat{V}_O$

Both $\hat{V}_N$ and $\hat{V}_O$ are L_2 consistent under certain moment and mixing conditions by, e.g., Carlstein (1986), Chien, Goldsman, and Melamed (1996), or under a strong approximation assumption, by Damerji (1994). This raises the natural question as to which is preferable. It can be seen that the two have approximately the same expectation so we compare their variances. $\hat{V}_O$ employs many more summands than

$\hat{V}_N$ so it may hope to be less variable, but the replicates contain a great deal of redundancy so it is not clear it is at all advantageous. It is difficult to compare for a general statistic s_n, so we now assume that $s_n = \overline{X}_n$, the sample mean.

Meketon and Schmeiser (1984) show the following:

Result

Assume that $b \to \infty$ and $b/n \to 0$.

Then:

$$\frac{Var(\hat{V}_O)}{Var(\hat{V}_N)} \to 2/3. \tag{1}$$

Thus, if one can afford the additional computational burden, it is preferable to use all possible batches of length b. This result has also been obtained by Künsch (1989) in his study of the "blockwise bootstrap". His basic idea is to choose k batches of length b from the set of all possible batches of length b, "glue" them together, and calculate s_n^*, say. If this is done B times then the "block bootstrap" estimator of $Var(s_n)$ is simply the (unstandardized) sample variance of the B values of s_n^*.

The above Result was actually obtained by Cox and Lewis (1961) in the case of estimating the intensity of a Poisson process. In fact, their formulation allows for any amount of overlap between the batches. In particular, let $\hat{V}_H$ and $\hat{V}_T$ denote the variance estimators with half and 3/4 overlap, respectively. Their results show that

$$\frac{Var(\hat{V}_O)}{Var(\hat{V}_H)} \to 8/9$$

and that

$$\frac{Var(\hat{V}_O)}{Var(\hat{V}_T)} \to 32/33.$$

Note, for example, if $n = 2000$ and $b = 100$ then the number of batches employed in $\hat{V}_N, \hat{V}_H, \hat{V}_T, \hat{V}_O$ is 20, 39, 77, and 1901, respectively. Thus, the "partially" overlapping $\hat{V}_H$ and $\hat{V}_T$ may be attractive alternatives to $\hat{V}_O$ if computations are costly. On the other hand only $\hat{V}_O$ is easily implemented for arbitrary values of b and n. Further, as shown by Song and Schmeiser (1995, Table 1) the performance of $\hat{V}_O$ is less sensitive to the exact choice of batch size than is $\hat{V}_N$. For these reasons, and due to its superior stability we will focus on $\hat{V}_O$ in the sequel.

2.2 Asymptotically Optimal Choice of Batch Size

Previous asymptotic results (e.g., L_2 consistency) have been important in justifying the use of batching for variance estimation. They ensure asymptotic validity of batching for a broad range of batch sizes. Nevertheless, for any given simulation an important practical question is: How should we select batch size, b? In the sequel we define the optimal b for an estimator $\hat{V}$ to be the one that minimizes $MSE(\hat{V}) := Bias^2(\hat{V}) + Var(\hat{V})$. In this section we discuss two results that give the order of magnitude of b as a function of n.

Carlstein (1986) adresses the question of batch size by examining the statistic $\overline{X}_n$ in the special case where the output sequence is generated by the AR(1) process:

$$X_i = \mu + \rho(X_{i-1} - \mu) + \epsilon_i, \tag{2}$$

where ϵ_i's are independent standard normal random variables and $|\rho| < 1$. He shows that in this situation:

$$Bias(\hat{V}_N) = \frac{-2\rho}{(1-\rho)^3(1+\rho)}(1/b) + o(1/b),$$

$$Var(\hat{V}_N) = \frac{2}{(1-\rho)^4}(b/n) + o(b/n),$$

and thus that the asymptotically optimal b is:

$$b_{opt} = (\frac{2\rho}{1-\rho^2})^{2/3} n^{1/3}. \tag{3}$$

This shows that larger batch sizes are needed for stronger correlations in the sequence, which is intuitively reasonable. Equation (3) also suggests a method for obtaining b in any given situation. Assume (temporarily) that the sequence is generated by an AR(1) process, estimate ρ (e.g., by Least Squares), and plug the resulting $\hat{\rho}$ into equation (3). For the estimator based on overlapping batches, $\hat{V}_O$, the constant 2 in the expression for $Var(\hat{V}_N)$ is replaced by 4/3 (by equation (1)) and thus in (3) the constant 2 is replaced by $6^{1/2}$. Thus, the reduced variance of $\hat{V}_O$ allows for a larger batch size to attempt to reduce bias.

Song and Schmeiser (1988, 1995) discuss a more general result for various estimators:

Assume that

$$Bias(\hat{V}) = -(1/b)c_b\gamma_1 + o(1/b),$$

and that

$$Var(\hat{V}) = (b/n)c_v\gamma_0^2 + o(b/n).$$

Then:

$$b_{opt} := (\frac{2c_b^2\gamma_1^2}{c_v\gamma_0^2})^{1/3} n^{1/3} =: Cn^{1/3}, \qquad (4)$$

where c_b, c_v are constants depending on the overlapping scheme, for example for $\hat{V}_O$ we have $c_b = 1$ and $c_v = 4/3$, while for $\hat{V}_N$ we have $c_b = 1$ and $c_v = 2$. The constants $\gamma_j, j = 0, 1$, are defined by $\gamma_j = \sum_{i=-\infty}^{\infty} i^j\gamma(i)$, so that, in particular, γ_0 is equal to V itself.

Song (1996) has considered estimating the ratio $(\gamma_1/\gamma_0)^2$ directly and plugging the resulting estimate into the middle expression in (4) using the known values of c_b and c_v. She shows that this can be done effectively in various settings. The results, however, are somewhat situation specific. In each case considered the user needs to analytically derive the constants c_b and c_v. For example, if the user wants to use the estimator $\hat{V}_H$ or $\hat{V}_T$ (described in Section 2.1) the constants c_b and c_v must be explicitly derived. Further, the results only naturally apply when the statistic of interest is the sample mean. Part of the beauty of the batch means methodology is that the method is generally applicable without regards to the statistic being computed (e.g., sample mean, sample quantile, linear combination of order statistics).

In the following section we propose a method to estimate batch size which is omnibus. That is, only the right hand side of equation (4) will be used. Thus, no knowledge of the constants in the middle expression of (4) is necessary. Further the method is quite general and can be applied in a natural way to arbitrarily complicated statistics.

3 A MODEL FREE DATA BASED CHOICE OF BATCH SIZE

The statistics and simulation communities have long sought an effective general method of determining an appropriate batch size without making assumptions on the mechanism generating the output (e.g., AR(1)). Noting just two comments from the recent literature, Sargent, Kang, and Goldsman (1992) say that "a good batch size estimation procedure would be of tremendous importance" while Damerji (1994) says "batch-size selection is still an unresolved problem".

We will focus on the estimator $\hat{V}_O$ although the proposed method is generally applicable to the other batch variance estimators discussed. We consider only $s_n = \overline{X}_n$, although the method is applicable to other "mean-like" statistics. More precisely, the method is appropriate whenever (4) holds for some (usually unknown) constant C. The basic idea is to empirically estimate the best b for a sequence of smaller length, m, and then extrapolate to obtain the best b for the entire sequence of length n. The basis of this is as follows:

Note from the result in Section 2.2, (equation (4)) that the asymptotically optimal b is of the form $b_n = Cn^{1/3}$ where the constant C depends only on the underlying process and not on the simulation length n. For any shorter sequence of length m, say, we have $b_m = Cm^{1/3}$ and thus $b_n = (n/m)^{1/3}b_m$. If we can estimate b_m empirically by $\hat{b}_m$ then our estimate of b_n will simply be $\hat{b}_n = (n/m)^{1/3}\hat{b}_m$. It turns out that a natural empirical estimate of b_m exists. Towards this end we give:

An Algorithm for Estimating b_n

1) Choose a pilot value for b, $b = b^*$, say, and calculate $\hat{V}_O$ using $b = b^*$.

2) For some m, consider $X_m^i, i = 1, ..., (n-m+1)$, all possible series of length m. For the i'th series of length m, let $\hat{V}_{m^*}^i$ denote the batch variance estimator computed from the series X_m^i using a batch size of m^*, and define:

$$\hat{b}_m = \text{argmin}_{m^*} \sum_{i=1}^{n-m+1} (\hat{V}_{m^*}^i - \hat{V}_O)^2/(n-m+1).$$

This is the empirical estimate of the b_m that minimizes $MSE(\hat{V}_O)$ for a sequence of length m.

3) Compute $\hat{b}_n = (n/m)^{1/3}\hat{b}_m$.

The algorithm could be iterated if desired. In other words, we could set b^* in step 1) equal to $\hat{b}_n$ in step 3) and again obtain $\hat{b}_m$ in step 2, and thus a new $\hat{b}_n$. In the examples to follow the algorithm converged in the single iteration approximately 90 percent of the time, and for this reason no iteration was performed. We note that a similar algorithm for the purpose of estimating the distribution function or the bias of an estimator has been suggested by Hall and Jing (1994). In the Appendix we outline an argument justifying the algorithm.

Note that, as discussed previously, although we focus on $\overline{X}_n$ the algorithm should be effective for any statistic such that equation (4) holds for some constants, i.e., b_{opt} is asymptotically a constant times $n^{1/3}$.

We perform three experiments to study the effectiveness of the proposed algorithm. The first generates output from an autoregressive model, the second from a moving average model, and the third considers waiting times in an M/M/1 queue.

3.1 Experiment 1: Autoregressive Output

The model generating the output sequence is an AR(1) sequence as described in Equation (2), where without loss of generality we take $\mu = 0$ throughout. From Equation (3) we know the asymptotically optimal b (recall that $6^{1/2}$ replaces the constant 2 for the estimator $\hat{V}_O$), and we take this to be the correct optimal batch length. It can be shown that in all cases considered the asymptotically optimal batch length coincides approximately with finite sample optimality. The optimal values of b for each setting considered are given in Table 1.

Table 1: Optimal Batch Sizes for the AR(1) Process

	ρ		
n	.2	.5	.8
200	4	8	18
1000	6	14	31

If there is any discrepancy between finite sample and asymptotically optimal batch sizes it is that the asymptotic values are slightly too large. For example, the biggest discrepancy occurs for $n = 200, \rho = .8$ where the true optimal value is $b = 16$. Using the asymptotic values for comparison can only hurt the algorithm because we will see that the algorithm typically underestimates the asymptotically optimal batch size. Thus, the asymptotic values give a conservative benchmark by which to judge the algorithm.

100 output sequences of length $n = 200$ were generated from equation (2) with ρ =.2. In each case the pilot value in Step 1 was b^* =10 with m=10. For each sequence the algorithm was carried out (without iteration). In this case $(n/m)^{1/3} \simeq 2.714$ so we take $\hat{b}_n$ to be the closest integer to $2.714\hat{b}_m$. In this case the optimal value is $b = 4$. In the 100 trials the algorithm chose $\hat{b}_n = 3$ ($\hat{b}_m = 1$) in 88% of the trials and $\hat{b}_n = 5$ ($\hat{b}_m = 2$) in the remaining 12% of the trials. The average b chosen is thus approximately 3.24 with an estimated standard error of $(.4224/100)^{1/2} = .06$.

This entire procedure was carried out for the two simulation lengths, $n = 200$ and $n = 1000$, for each of three different strengths of correlation, $\rho = .2, \rho = .5$, and $\rho = .8$, for each of two different pilot values $b^* = n/20$ and $b^* = n/10$. These are reasonable pilot values, and in accordance with those suggested by Schmeiser (1982). In all cases $m = n/20$, where m is as defined in step two of the algorithm. This would give 20 nonoverlapping replicates from which we could empirically estimate the optimal b_m. Because we use overlapping batches, however, the effective number is (from equation (1)) approximately 20(3/2)=30. This choice of m is clearly not the best in all cases, but it seems to work reasonably well in both the $n = 200$ and $n = 1000$ cases in this and the following examples.

Given in Table 2 are the average results and the estimated standard error of the average for each setting.

Table 2: Batch Size Determination for AR(1) Output

Setting	pilot b^*	$\overline{b}$	st. error
n=200, $\rho = .2$	10	3.2	.06
optimal b=4	20	3.5	.10
n=200, $\rho = .5$	10	5.1	.12
optimal b=8	20	5.5	.20
n=200, $\rho = .8$	10	8.1	.17
optimal b=18	20	11.5	.32
n=1000, $\rho = .2$	50	6.7	.34
optimal b=6	100	7.0	.45
n=1000, $\rho = .5$	50	12.0	.46
optimal b=14	100	12.8	.53
n=1000, $\rho = .8$	50	23.1	.43
optimal b=31	100	24.2	.84

The results indicate that the algorithm is a promising method of selecting batch length. The algorithm performs exceptionally well when the dependence is weak, $\rho = .2$, reasonably well for moderate dependence, $\rho = .5$, and has the most trouble with strong dependence, $\rho = .8$. In general, the algorithm performs better for the larger simulation length, $n = 1000$. In all but one case the larger pilot value brings the algorithm closer to the correct optimal batch length (on average), but at the price of greater variability. This is reasonable, as in Step 2) of the algorithm, the estimated MSE is closest to the true when $\hat{V}_O$ is closest to V. The larger pilot value enables $\hat{V}_O$ to be closer to V (on average) but at the cost of increased variability. It seems that results will be bad if the pilot value is much smaller than the true optimal b.

In order to compare the effectiveness of the algorithm, across ρ and n, consider the measure "effective sample size", the effective number of independent observations in the simulation. This is approximately $n^* := Var(X)/V$ which is equal to $n(1-\rho)/(1+\rho)$ in the autoregressive case. Given in Table 3 is the effective sample size and the ratio, R, of the average batch length chosen by the algorithm to the optimal batch length, for each setting (the average value is based on the $b^* = n/10$ pilot value in each case).

We see that, uniformly, the higher the effective sample size, the higher the ratio R. For the most part, the algorithm works better for the larger effective sample sizes, as is only natural. It is not clear why for the $n = 1000, \rho = .2$ case the estimated batch length is too big. It turns out that the distribution of the batch length is skewed right in this case, and the median of the 100 values is 5, which is less than the optimal value of $b = 6$. In general, the distribution of the estimated batch length is skewed to the right. Thus, the estimated batch length shares two features common to the variance estimator itself. Namely, both are negatively biased for the object of estimation, and both have skewed right distributions. The latter is not suprising as both estimators are non-negative.

Perhaps a better gauge as to how well the algorithm performs is to compare the Mean Square Errors (MSE's) of the variance estimators based on the optimal batch length to those based on the mean batch length chosen by the algorithm. We use the results using the larger pilot value b^* from Table 2. Given in Table 4 are the average, variance, and MSE's of the variance estimators for each of the settings in Table 2. The number of Monte Carlo iterations for each setting was 10000, the results differ slightly from the predicted asymptotic values. The target value $V = (1-\rho)^{-2}$ is equal to 1.56, 4, and 25, for $\rho = .2$, $\rho = .5$, and $\rho = .8$, respectively. We could use the finite sample estimates of variance, but this would have very little effect on the relative performance of the optimal batch length to the mean batch length from the algorithm, so we use the asymptotic target V for comparison.

It can be seen that there is little difference between the performance of the optimal batch length and the mean batch length chosen by the algorithm. The greatest discrepancy occurs when $\rho = .8$. In this case the ratio of MSE's is approximately 93%. This is much higher than the ratio of corresponding batch sizes in Table 3. In fact, in all situations considered, the ratio of the MSE's is closer to 1 than the ratio of the batch sizes themselves. We see that in all cases the algorithm, on average, gives MSE's which are competitive with the best possible.

3.2 Experiment 2: Moving Average Output

In this example we generate output from the moving average model:

$$X_i = \mu + \theta\epsilon_{i-1} + \epsilon_i,$$

where the ϵ_i's are independent standard normal random variables and $\theta \in R^1$. In this case $\gamma(0) := Var(X) = 1+\theta^2, \gamma(1) = \theta$, and thus $V = 1+\theta^2+2\theta = (1+\theta)^2$. Hence, the constants in equation (4) are as follows: $c_b = 1, c_v = 4/3, \gamma_0 = V = (1+\theta)^2$, and $\gamma_1 = 2\theta$. Thus, $b_{opt} = (6\theta^2/(1+\theta)^4)^{1/3}n^{1/3}$. Given in Table 5 are the optimal values for each of the six situations we will consider.

We use the same two pilot values as in Section 3.1 and once again all average batch lengths and standard errors are based on 100 Monte Carlo runs for each setting. The results are given in Table 6.

The same conclusions reached from the autoregessive process hold here as well. For fixed simulation length the method performs best for weaker correlations. For fixed correlation, the method seems to perform better for the longer simulation length, $n = 1000$, although this effect is partially obscured by the rounding to integer values.

3.3 Experiment 3: Output from an M/M/1 Queue

In this example we study the algorithm in the setting of a $M/M/1$ queue. For $i = 1, 2, ..., n$, let A_i denote the waiting time between the $i-1$'th and i'th customer, S_i the service time for the i'th customer, and let $W_i = Max\{0, W_{i-1} + S_{i-1} + A_i\}$ denote Lindley's equation, Grimmett and Stirzaker (1985), the amount of time spent waiting by the i'th customer.

Assume that A_i and S_i are independendent exponential random variables with respective means $(1/\lambda)$ and $(1/\nu)$. We consider the centered sequence $X_i := W_i - E(W_i) = W_i - \lambda/(\nu(\nu-\lambda))$, see Song and Schmeiser (1988, 1995) for further descriptions of this process.

We consider the case where $\lambda = 1$ and $\nu = 2$, for the same two simulation lengths and pilot values used in Sections 3.1 and 3.2. This gives a moderate traffic density of $\lambda/\nu = 1/2$.

We compare the MSE obtained using the optimal batch size and the MSE obtained from the mean batch length chosen by the algorithm. Song and Schmeiser (1995) point out that for this process the asymptotic and finite sample optimal batch sizes can differ considerably, even for long simulations. For this reason we determine the optimal batch size and optimal MSE for each sample size considered via Monte Carlo experiments.

From 100,000 Monte Carlo iterations we find that the target values are $V = 6.74$ for $n = 200$ and $V = 7.16$ for $n = 1000$. For $n = 200$, we find that the optimal batch size is $b_n = 5$, with an associated MSE of 25.1. This last quantity was estimated from 20,000 independent computations of $\hat{V}$ using a batch size of $b_n = 5$. Similarly, we find that the optimal batch size when $n = 1000$ is $b_n = 11$, with an associated MSE of 16.3. Table 7 gives the optimal batch size for each setting and the average chosen by the algorithm with associated standard errors. Table 8 gives the optimal MSE and the MSE obtained from the average batch size chosen by the algorithm for each of the two pilot values $b^* = n/10$ and $b^* = n/20$.

We see that for both pilot values the MSE from the typical batch size chosen by the algorithm competes relatively well with the optimal MSE.

4 CONCLUSION

We have presented a data based method for determining batch length in batch means variance estimation procedures. Unlike earlier methods (e.g., Carlstein (1986)) the method does not rely on knowledge of the underlying dependence structure to estimate batch length. Unlike the method of Song (1996), the method does not require analytical calculations for each different estimator (e.g., partially overlapping batches) and for each different statistic. We have seen that for reasonable simulation lengths, and for moderate dependence within the output sequence, the method performs relatively well.

The method depends on the choice of two parameters: m and b^*. The former determines the number of replicates for the empirical estimate of the mean squared error of batch variance estimators of (the standardized) $Var(\overline{X}_m)$, while the latter is the pilot estimator of optimal batch length for estimating (the standardized) $Var(\overline{X}_n)$. Choosing $m = n/20$ gives effectively 30 replicate experiments which has worked reasonably well in all cases considered. Overall, it seems appropriate to choose a relatively large pilot value b^*, e.g., $n/20$ or $n/10$, where n is the simulation length. Choosing a pilot value that is much too small can lead to a large underestimate of batch size, while choosing one that is too large seems to lead to at worst a slight overestimate of batch size, at the cost however of increased variability of estimated batch size.

We see that the MSE's using the algorithm track the MSE from the optimal batch size quite well in the three settings considered: MA(1) output, AR(1) output, and output from a M/M/1 queue. We have used a simple form of the batch estimator based on all possible overlapping series. Different scalings for the estimator may improve results. The method logically applies equally well to other estimators than those considered here. For example, nonoverlapping, partially overlapping, or spaced batches. For nonoverlapping and spaced batches, however, much longer simulation runs would be required in order to get a reasonable number of valid replicate experiments.

APPENDIX: An Exploration of Consistency of the Algorithm

We explore conditions under which $\hat{b}_n/b_n \xrightarrow{p} 1$. Recall that $\hat{b}_n = (n/m)^{1/3}\hat{b}_m, b_n/Cn^{1/3} \to 1$ as $n \to \infty$, and $b_m/Cm^{1/3} \to 1$ as $m \to \infty$, and thus we only need that $\hat{b}_m/b_m \xrightarrow{p} 1$.

$b_m := \text{argmin}_{m^*} E(\hat{V}_{m^*} - V)^2$ and we have defined $\hat{b}_m = \text{argmin}_{m^*} \sum_{i=1}^{n-m+1} (\hat{V}^i_{m^*} - \hat{V}_O)^2/(n-m+1)$.

We will show that under reasonable conditions $\sum_{i=1}^{n-m+1} (\hat{V}^i_{m^*} - \hat{V}_O)^2/(n-m+1) - E(\hat{V}_{m^*} - V)^2 \xrightarrow{p} 0$ as $m \to \infty, m/n \to 0$.

First: $(\sum_{i=1}^{n-m+1} (\hat{V}^i_{m^*} - \hat{V}_O)^2/(n - m+1) \sum_{i=1}^{n-m+1} (\hat{V}^i_{m^*} - V)^2/(n-m+1) = 2(\hat{V}_O - V)\sum_{i=1}^{n-m+1} \hat{V}^i_{m^*}/(n-m+1) + (\hat{V}_O^2 - V^2)$.

$\hat{V}_O \xrightarrow{p} V$ implies that $\hat{V}_O^2 - V^2 \xrightarrow{p} 0$. $\sum_{i=1}^{n-m+1} \hat{V}^i_{m^*}/(n-m+1) = O_p(1)$ if the variance estimator has uniformly bounded second moment (which holds under the usual moment and mixing conditions that gaurantee L_2 consistency of the variance estimator). Together, these two statements imply:

$$\sum_{i=1}^{n-m+1} \{(\hat{V}^i_{m^*} - \hat{V}_O)^2 - (\hat{V}^i_{m^*} - V)^2)\}/(n-m+1) \xrightarrow{p} 0.$$

Further, $\sum_{i=1}^{n-m+1} (\hat{V}^i_{m^*} - V)^2/(n - m + 1) - E(\hat{V}_{m^*} - V)^2 =: \sum_{i=1}^{n-m+1} t^i_{m^*,m}/(n-m+1)$ where $E(t^i_{m^*,m}) = 0$. This average tends to 0 in probability if each summand is uniformly integrable, i.e., if $E(\hat{V}_n - V)^{2+\delta}$ is uniformly bounded for some $\delta > 0$ and the usual polynomial decay of mixing coefficients holds.

Thus far we have, $\sum_{i=1}^{n-m+1} (\hat{V}^i_{m^*} - \hat{V}_O)^2/(n-m+1) - E(\hat{V}_{m^*} - V)^2 \xrightarrow{p} 0$, as $m \to \infty, m/n \to 0$.

In order to conclude that $\hat{b}_m/b_m \xrightarrow{p} 1$ we would need to apply the following Lemma from Andersen and Gill (1982).

Lemma :

Let $f_1, f_2, \ldots,$ be a sequence of random convex functions such that for each $x \in R, f_n(x) \xrightarrow{p} f(x)$. If $X_n := \text{argmin}_x f_n(x)$ and $x_* := \text{argmin}_x f(x)$ exist uniquely, then $X_n \xrightarrow{p} x_*$.

Unfortunately, we cannot use this Lemma because $f_n := \sum_{i=1}^{n-m+1} (\hat{V}^i_{m^*} - \hat{V}_O)^2/(n-m+1)$ may not be, in general, convex in m^*, although empirical evidence suggests it is "usually so", particularly for large n.

Nevertheless, we explore conditions under which the limit $f := E(\hat{V}_{m^*} - V)^2$ is convex in m^* for fixed m, n. From the Lemma this is essentially a necessary condition. From the results that led to (4) in Section 2.2 we have that $E(\hat{V}_{m^*} - V)^2$ is asymptotically equivalent to:

$$(c_b^2\gamma_1^2)/m_*^2 + (m_*/m)c_v\gamma_0^2.$$

The second derivative of this last expression with respect to batch size m_* is equal to $6c_b^2\gamma_1^2/m_*^4$ which is positive for all m_*. Thus $E(\hat{V}_{m^*} - V)^2$ is convex.

The above arguments give some theoretical basis to indicate that the algorithm performs well for large samples. The finite sample verification of this is indicated in the simulations in Section 3.

ACKNOWLEDGMENT

The author sincerely thanks David Goldsman, Barry Nelson, Bruce Schmeiser, and Wheyming Tina Song for discussions held at the 1996 Winter Simulation Conference.

REFERENCES

Andersen, P.K. and Gill, R.D. 1982. Cox's regression model for counting processes: a large sample study. *Annals of Statistics* 10:1100–1120.

Bratley, P., Fox, B.L., and Schrage, L.E. 1987. *A guide to simulation.* Springer Verlag, N.Y.

Carlstein, E. 1986. The use of subseries values for estimating the variance of a general statistic from a stationary sequence. *Annals of Statistics* 14:1171–1179.

Chien, C., Goldsman, D., and Melamed, B. 1996 (to appear). Large-sample results for batch means. *Management Science*.

Cox, D.R. and Lewis, P. 1961. *The statistical analysis of series of events.* London, Methuen.

Damerji, H. 1994. Mean-square consistency of the variance estimator in steady-state simulation output analysis. *Operations Research* 43:282–291.

Fishman, G.S. 1978. *Principles of Discrete Event Simulation.* John Wiley & Sons, N.Y.

Fox, B.L., Goldsman, D, and Swain, J.J. 1991. Spaced batch means. *O.R. Letts.*. 10:255–263.

Gastwirth, J.L. and Rubin, H. 1975. The asymptotic distribution theory of the empiric c.d.f. for mixing stochastic processes. *Annals of Statistics* 3:809–824.

Glynn, P.W. and Inglehart, D.L. 1990. Simulation output analysis using standardized time series. *Mathematics of Operations Research* 15:1–16.

Grimmett, G.R. and Stirzaker, D.R. 1985. *Probability and random processes.* Oxford University Press, Oxford.

Hall, P. and Jing, B.Y. 1994. On sample re-use methods for dependent data. *Research Report*, #SR8-94, Centre for Math. and its Appl., Australian National Univ., Canberra.

Künsch, H. 1989. The jackknife and bootstrap for general stationary observations. *Annals of Statistics* 17:1217–1241.

Meketon, M.S. and Schmeiser, B. 1984. Overlapping batch means: something for nothing? In *Proc. of the Winter Simulation Conference*, 227–230.

Pedrosa, A.C. and Schmeiser, B.W. 1993. Asymptotic and finite-sample correlations between OBM estimators. In *Proc. of the 1993 Winter Simulation Conference*, 481–488.

Sargent, R.G., Kang, K., and Goldsman, D. 1992. An investigation of finite-sample behavior of confidence interval estimators. *Operations Research* 40:898–913.

Schmeiser, B. 1982. Batch size effects in the analysis of simulation output. *Operations Research* 30:556–568.

Schruben, L. 1983. Confidence interval estimation using standardized time series. *Operations Research* 31:1090–1108.

Serfling, R.J. 1980. *Approximation theorems of mathematical statistics.* Wiley and Sons, N.Y.

Song, W.T. and Schmeiser, B. 1988. Estimating standard errors: empirical behavior of asymptotic MSE-optimal batch sizes. In *Comp. Science and Stat.: Proc. of the 20th Symposimum on the Interface*, 575–580.

Song, W.T. and Schmeiser, B. 1995. Optimal mean-squared-error batch sizes. *Management Science* 41:110–123.

Song, W.T. 1996 (to appear). On the estimation of optimal batch sizes in the analysis of simulation output. *European Journal of Operational Research.*

Table 3: Effective Sample Sizes and Effectiveness of Algorithm

n	ρ	n^*	R
200	.2	133	.85
	.5	67	.69
	.8	22	.64
1000	.2	667	1.16
	.5	333	.91
	.8	111	.78

Table 4: Comparison of MSE's from optimal batch size and algorithm

Setting	batch size	Mean	Var.	MSE
n=200, $\rho = .2$	b_{opt}=4	1.39	.061	.090
	$\bar{b}$=4	1.39	.061	.090
n=200, $\rho = .5$	b_{opt}=8	3.33	.780	1.23
	$\bar{b}$=6	3.12	.531	1.30
n=200, $\rho = .8$	b_{opt}=18	18.9	66.1	103
	$\bar{b}$=12	16.3	35.9	111
n=1000, $\rho = .2$	b_{opt}=6	1.45	.019	.031
	$\bar{b}$=7	1.47	.022	.030
n=1000, $\rho = .5$	b_{opt}=14	3.62	.281	.425
	$\bar{b}$=13	3.56	.237	.431
n=1000, $\rho = .8$	b_{opt}=31	21.5	23.6	35.9
	$\bar{b}$=24	20.4	17.2	38.4

Table 5: Optimal Batch Sizes for the MA(1) Output

	θ		
n	.2	.5	1
200	3	4	5
1000	5	7	9

Table 6: Batch Size Determinination for MA(1) Output

Setting	pilot b^*	$\bar{b}$	st. error
n=200, $\theta = .2$	10	3.2	.06
optimal b=3	20	3.4	.09
n=200, $\theta = .5$	10	3.7	.10
optimal b=4	20	3.9	.12
n=200, $\theta = 1$	10	4.1	.10
optimal b=5	20	4.2	.15
n=1000, $\theta = .2$	50	5.6	.25
optimal b=5	100	5.9	.34
n=1000, $\theta = .5$	50	7.2	.35
optimal b=7	100	7.5	.42
n=1000, $\theta = 1$	50	7.6	.37
optimal b=9	100	8.0	.44

Table 7: Batch Size Determinination for M/M/1 Queue

Setting	pilot b^*	$\bar{b}$	st. error
n=200	10	4.93	.29
optimal b=5	20	6.38	.48
n=1000	50	9.11	.34
optimal b=11	100	10.9	.86

Table 8: Comparison of MSE's from M/M/1 Queue

Setting	batch size	MSE
n=200	b_{opt}=5	25.1
pilot $b^* = 10$	$\bar{b}$=5	25.1
pilot $b^* = 20$	$\bar{b}$=6	25.5
n=1000	b_{opt}=11	16.3
pilot $b^* = 50$	$\bar{b}$=9	16.9
pilot $b^* = 100$	$\bar{b}$=11	16.3