LABATCH.2 for Analyzing Sample Path Data

George S. Fishman

Department of Operations Research
University of North Carolina
Chapel Hill, NC 27599

UNC/OR TR-97/04

December 1997

Preliminary Version

Abstract

LABATCH.2 is a collection of computer programs available in C, FORTRAN, and SIMSCRIPT II.5 that perform statistical analyses on sample sequences collected on strictly stationary stochastic processes. It has been designed to make its implementation easy for potential users. It may be invoked in-line each time a data vector is generated within an executing program (e.g., simulation), or it may take its input from a stored data file. For sample path length t, LABATCH.2 takes $O(t)$ computing time and $O(\log_2 t)$ space. LABATCH.2 is a revised version of the LABATCH statistical analysis package and is considerably more user-friendly and versatile than the original.

In addition to computing a sample average for each series, LABATCH.2 provides an asymptotically valid confidence interval for assessing how well the sample average approximates the true unknown mean of the series. The confidence interval uses a consistent estimator of the variance of the sample mean computed by a dynamic version of the batch means method. As part of its output for each series, LABATCH.2 produces a sequence of interim estimates of this variance as the sample path length evolves, thus enabling a user to assess the extent to which systematic error in the latest variance estimate, due to ignoring correlation between batches, has dissipated as batch size grows with increasing sample path length. These tableaus also produce data that allow a user to assess the extent to which the sample average for each series is free of the initial conditions that prevailed when data collection began.

LABATCH.2 offers the user an interactive option that displays interim variance estimates for up to seven series on the screen at each update. The user may then instruct LABATCH.2 to continue execution until the next update or, if the variance estimates are regarded as sufficiently small, may instruct LABATCH.2 to terminate statistical analysis and write the final results to a file. This option allows the user to revise downward the sample path length, originally specified at the beginning of execution, thus saving computing time.

Section 1 uses output tableaus and graphs based on these tableaus to illustrate the statistical information that LABATCH.2 provides. It also describes its user-specified input that controls execution of LABATCH.2. Section 2 concisely describes the estimating techniques and the method of determining batch size as the sample path evolves. For a user-specified sample path length t, the section describes how LABATCH.2 selects its initial batch size and number of batches to maximize the number of observations used to estimate the variance on final review. Under relatively weak constraints, this number always exceeds $.888t$.

Section 3 illustrates how a graphical display of LABATCH.2 output enables a user to assess whether or not the program's final output is free of the influence of the conditions prevailing at the time that data collection begins. Section 4 describes several major features of the LABATCH.2 programs and how one can take advantage of them in practice.

Introduction

To be meaningful, each sample average generated by a discrete-event simulation with random input needs to be accompanied by an indicator of how well it approximates the parameter for which it serves as a point estimate. A sample average is subject to two sources of error; a systematic bias induced by starting the simulation run in an arbitrarily chosen, rather than an equilibrium, state and a random error arising from sampling variation along the sample path. Although both errors become vanishingly small with increasing sample path length for most simulations, the inevitability of always working with finite path lengths leaves no alternative for meaningful interpretation of the sample average but to acknowledge the presence of error and to assess its relative importance.

Since the typical simulation user's interest rarely lies in statistical analysis, only measures of assessment automatically generated during or at the completion of a simulation run can be expected to attract her/his attention. LABATCH.2 is a collection of computer programs designed to provide these measures. In particular, it performs statistical analyses on sample sequences collected on strictly stationary stochastic processes and offers two modes of implementation. One integrates LABATCH.2 into an executing data-generating program (e.g., simulation) to analyze the evolving data on repeated subroutine calls; the other takes data from an existing file as input. In addition to minimizing user effort, the first option considerably reduces space requirements. It also allows user interaction with the executing program via screen displays of interim estimates. The second option permits statistical analysis of stored data, regardless of source and date of generation, thereby making LABATCH.2 applicable in a considerably wider range of data-generating environments.

LABATCH.2 is a revision of LABATCH (Fishman 1996, Fishman and Yarberry 1997) that considerably simplifies its implementation and use. The simplifications are its most attractive feature. A user merely inserts a single subroutine call statement in her/his main program and assigns values to the ten control arguments of the subroutine. Section 1 gives a brief overview of what LABATCH.2 does, the output it generates, and the values that the control arguments can assume. Section 2 describes the basis for the way in which LABATCH.2 goes about its analysis.

Sections 1 and 2 assume that data collection begins when the simulation has attained steady state and, hence, error due to initial conditions is absent. In usual practice, a simulation user begins a simulation in a convenient, but arbitrarily chosen, state. To compensate for this choice, data collection begins after a warm-up interval that the simulator regards as adequate to make induced systematic bias negligible. Since this judgment is subject to error, there remains a nagging doubt about the adequacy of the warm-up interval. Section 3 describes how the LABATCH.2 tableaus can also provide an assessment of how well the selected truncation point has dissipated the influence of initial conditions.

Section 4 discusses several major features of LABATCH.2, including on-line screen display of interim results. C, FORTRAN, and SIMSCRIPT II.5 implementations of LABATCH.2 are obtainable by anonoymous file transfer procedure (ftp) at ____________.

1 Overview

For each sample sequence, $X_1, \ldots, X_t$, in its input, LABATCH.2 computes as part of its output a sample average, $\bar{X}_t$, as an estimate of its true unknown mean, μ, and an asymptotically valid $100 \times (1-\delta)$ percent confidence interval for assessing how well $\bar{X}_t$ approximates μ. The confidence interval relies crucially on an estimate, $BW(L,B)$, of the *asymptotic variance*, $\sigma^2_\infty := \lim_{t\to\infty} t \ \mathrm{var}\bar{X}_t$, computed by the *batch means method*, where B denotes batch size and L, the number of batches. Since this variance estimate is also subject to *systematic error* as well as *random sampling error* and since the validity of the confidence interval depends on this systematic error being relatively negligible, LABATCH.2 also displays interim calculations of $BW(L,B)$ that allow a user to assess the extent to which systematic error remains in the final variance estimate used to compute the confidence interval for the mean μ. The ability to make this assessment with LABATCH.2 output is one of its most valuable assets.

In the present context, systematic error in the variance estimate can be present even for sample data free of bias due to initial conditions. In particular, it arises from neglecting all but the term proportional to t^{-1} in $\mathrm{var}\bar{X}_t$ and from ignoring the correlation between batch averages. To provide a basis for systematic error assessment, LABATCH.2 computes a sequence of estimates of σ^2_∞ based on data subsequences of increasing lengths $t_1 < t_2 < \cdots < t_{J(t)} \le t$, where $t_{i+1} = 2t_i$ for $i = 1, \ldots, J(t)-1$ and where the user-specified path length, t, determines t_1 and $J(t)$ (Sec. 2).

As illustration, consider a simulation of the M/M/1 queueing model with .90 interarrival rate and unit service rate. The simulation began in the steady state and terminated when customer $t = 10^7$ entered service. Figure 1a displays LABATCH.2–computed 99 percent confidence intervals for the mean waiting time in queue (Series 1) and for the probability that a customer waits (Series 2). The true values are 9 and .90 respectively.

For each series, Fig. 1b displays LABATCH.2–computed sequences of point and 99 percent interval estimates of μ and point estimates, $\sqrt{BW(L,B)}$, of σ_∞ as the batch size, B, and the number of batches, L, grow with sample path length according to the ABATCH rule (Sec. 2.3). As sample path length doubles ($t_{j+1} = 2t_j$), the ABATCH rule doubles B if a test of the hypothesis H, batch averages are independent, detects systematic error. If no systematic error is detected (H is accepted), the rule increases B by a factor of $\sqrt{2}$ approximately. This implies that either L remains constant on successive reviews or L also increases approximately by a factor of $\sqrt{2}$. The $\sqrt{BW(L,B)}$ sequences for each series in Fig. 1b reveal that systematic error in the example has become negligible for reviews $j \ge 11$.

Figure 2 graphically displays the point and interval estimates (cols. 5, 6, and 7) for μ for each series. The graphs provide a convenient way of assessing the accuracy of the sample averages at a glance. These and all other graphs in the paper were created using Mathematica Ⓣ applied to the LABATCH.2 output after deleting the final tableau (Fig. 1a) and all header and trailer information from the interim review tableaus (Fig. 1b). EXCEL Ⓣ and similar software could have been used as alternatives.

Figure 3 graphically displays $\sqrt{BW(L,B)}$ (col. 8 in Fig. 1b) as estimates of σ_∞ for each series. For Series 1, $\sigma_\infty = 189.3$ (Blomqvist 1967). As Sec. 2.3 explains in more detail, they suggest negligible systematic error for Series 1 for estimates computed on reviews $j \ge 11$

Figure 1: LABATCH.2 output for M/M/1 queueing simulation (Series 1: waiting time in queue, Series 2: 1:=wait, 0:= no wait)

(a)

Final Tableau

Mean Estimation
****
(t = 10000000)

Series	$\bar{X}$	Standard Error Sqrt[B*W(L,B)/t]	99.0% Confidence Interval Lower	Upper	(Upper-Lower)/|$\bar{X}$|
1	0.8949D+01	0.5681D-01	0.8802D+01	0.9097D+01	0.3290D-01
2	0.8995D+00	0.4200D-03	0.8984D+00	0.9006D+00	0.2173D-02

$\bar{X}$ is based on all t observations.
W(L,B) is based on first 91.75% of the t observations.

(b)

Interim Review Tableau

ABATCH Data Analysis for Series 1

Review	L*B	L	B	$\bar{X}$	99.0% Confidence Interval Lower	Upper	Sqrt[B*W(L,B)]	p-value
1	35	7	5	0.1660D+02	0.1379D+02	0.1940D+02	0.4475D+01	0.3804
2	70	10	7	0.1810D+02	0.1575D+02	0.2045D+02	0.6047D+01	0.1109
3	140	14	10	0.1914D+02	0.1693D+02	0.2135D+02	0.8680D+01	0.1524
4	280	20	14	0.1886D+02	0.1662D+02	0.2111D+02	0.1313D+02	0.0194
5	560	20	28	0.1111D+02	0.5748D+01	0.1646D+02	0.4432D+02	0.0000
6	1120	20	56	0.7254D+01	0.2741D+01	0.1177D+02	0.5278D+02	0.0001
7	2240	20	112	0.6706D+01	0.2960D+01	0.1045D+02	0.6196D+02	0.0005
8	4480	20	224	0.7556D+01	0.3996D+01	0.1112D+02	0.8328D+02	0.0358
9	8960	20	448	0.6747D+01	0.4111D+01	0.9383D+01	0.8721D+02	0.7293
10	17920	28	640	0.7817D+01	0.5478D+01	0.1016D+02	0.1130D+03	0.7095
11	35840	40	896	0.9513D+01	0.6280D+01	0.1275D+02	0.2260D+03	0.0655
12	71680	40	1792	0.9668D+01	0.6693D+01	0.1264D+02	0.2941D+03	0.8172
13	143360	56	2560	0.9073D+01	0.7449D+01	0.1070D+02	0.2304D+03	0.7200
14	286720	80	3584	0.8883D+01	0.8012D+01	0.9754D+01	0.1767D+03	0.2993
15	573440	112	5120	0.9248D+01	0.8511D+01	0.9985D+01	0.2128D+03	0.7163
16	1146880	160	7168	0.9126D+01	0.8653D+01	0.9600D+01	0.1945D+03	0.0993
17	2293760	160	14336	0.9138D+01	0.8809D+01	0.9468D+01	0.1913D+03	0.1852
18	4587520	224	20480	0.9032D+01	0.8805D+01	0.9259D+01	0.1874D+03	0.7882
19	9175040	320	28672	0.8949D+01	0.8802D+01	0.9097D+01	0.1796D+03	0.2362
If data are independent:								
	10000000	10000000	1	0.8949D+01	0.8941D+01	0.8957D+01	0.9821D+01	0.0000

0.10 significance level for independence testing.
Review 19 used the first 91.75% of the t observations for W(L,B).

Interim Review Tableau

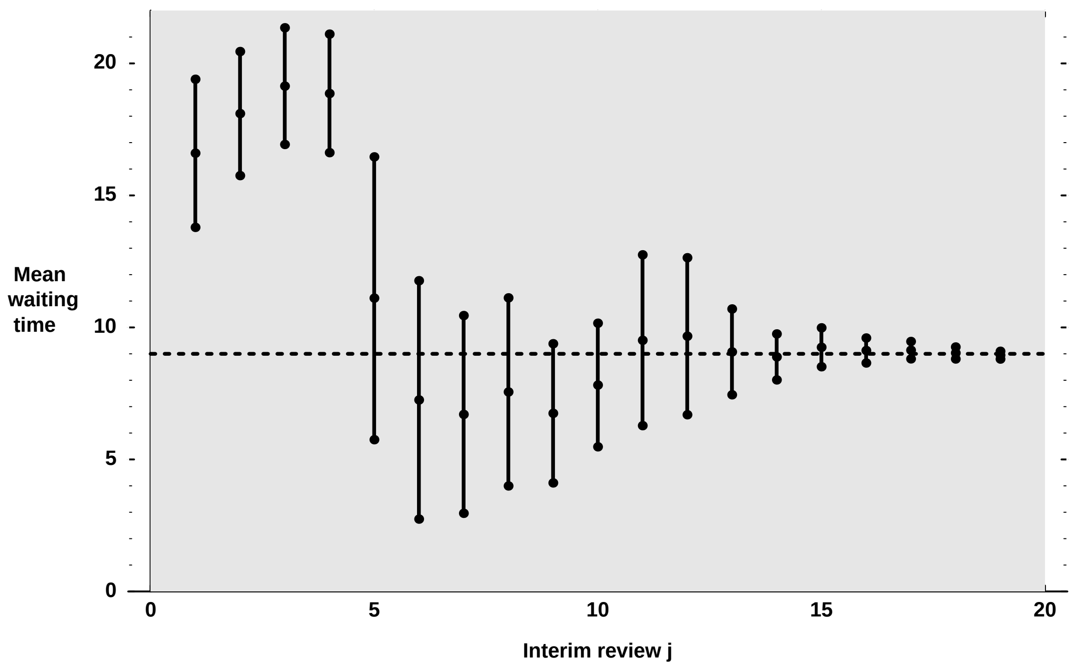
ABATCH Data Analysis for Series 2

Review	L*B	L	B	$\bar{X}$	99.0% Confidence Interval Lower	Upper	Sqrt[B*W(L,B)]	p-value
1	35	7	5	0.1000D+01	0.1000D+01	0.1000D+01	0.0000D+00	0.0000
2	70	7	10	0.1000D+01	0.1000D+01	0.1000D+01	0.0000D+00	0.0000
3	140	7	20	0.1000D+01	0.1000D+01	0.1000D+01	0.0000D+00	0.0000
4	280	7	40	0.1000D+01	0.1000D+01	0.1000D+01	0.0000D+00	0.0000
5	560	7	80	0.9250D+00	0.7838D+00	0.1066D+01	0.9014D+00	0.0067
6	1120	7	160	0.8812D+00	0.7697D+00	0.9928D+00	0.1007D+01	0.0140
7	2240	7	320	0.8862D+00	0.7503D+00	0.1022D+01	0.1734D+01	0.1002
8	4480	10	448	0.8830D+00	0.7975D+00	0.9686D+00	0.1762D+01	0.8214
9	8960	14	640	0.8770D+00	0.8360D+00	0.9180D+00	0.1289D+01	0.9589
10	17920	20	896	0.8888D+00	0.8607D+00	0.9169D+00	0.1316D+01	0.9648
11	35840	28	1280	0.9001D+00	0.8797D+00	0.9206D+00	0.1398D+01	0.3029
12	71680	40	1792	0.8973D+00	0.8813D+00	0.9133D+00	0.1581D+01	0.9235
13	143360	56	2560	0.8967D+00	0.8879D+00	0.9055D+00	0.1250D+01	0.6376
14	286720	80	3584	0.8976D+00	0.8916D+00	0.9036D+00	0.1220D+01	0.5008
15	573440	112	5120	0.9001D+00	0.8956D+00	0.9045D+00	0.1280D+01	0.1352
16	1146880	160	7168	0.8996D+00	0.8963D+00	0.9029D+00	0.1374D+01	0.5547
17	2293760	224	10240	0.8992D+00	0.8968D+00	0.9016D+00	0.1381D+01	0.6289
18	4587520	320	14336	0.8995D+00	0.8979D+00	0.9011D+00	0.1313D+01	0.7435
19	9175040	448	20480	0.8995D+00	0.8984D+00	0.9006D+00	0.1328D+01	0.6766
If data are independent:								
	10000000	10000000	1	0.8995D+00	0.8992D+00	0.8997D+00	0.3007D+00	0.0000

0.10 significance level for independence testing.
Review 19 used the first 91.75% of the t observations for W(L,B).

Figure 2: LABATCH.2 sample means and 99% confidence intervals for ABATCH rule; simulation starts in steady state

(a) Series 1



(b) Series 2

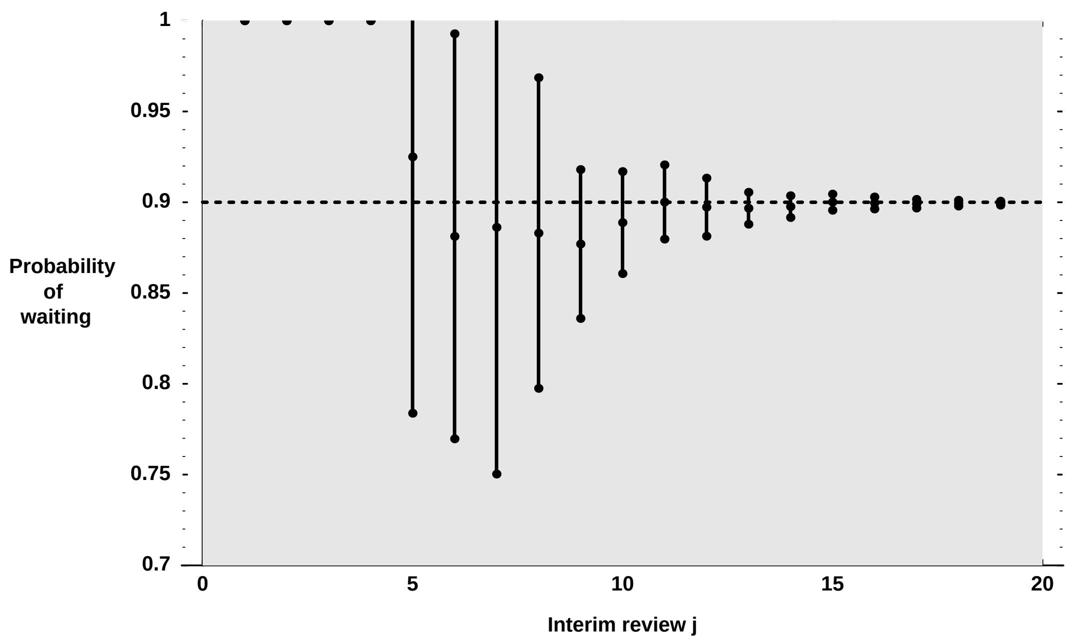
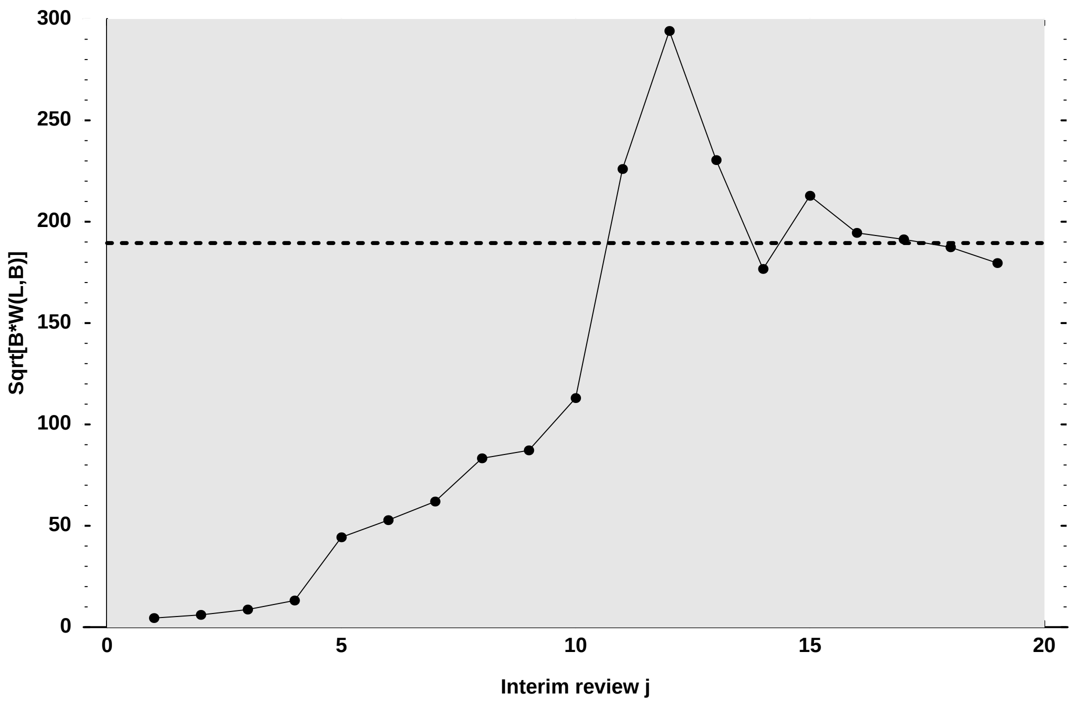
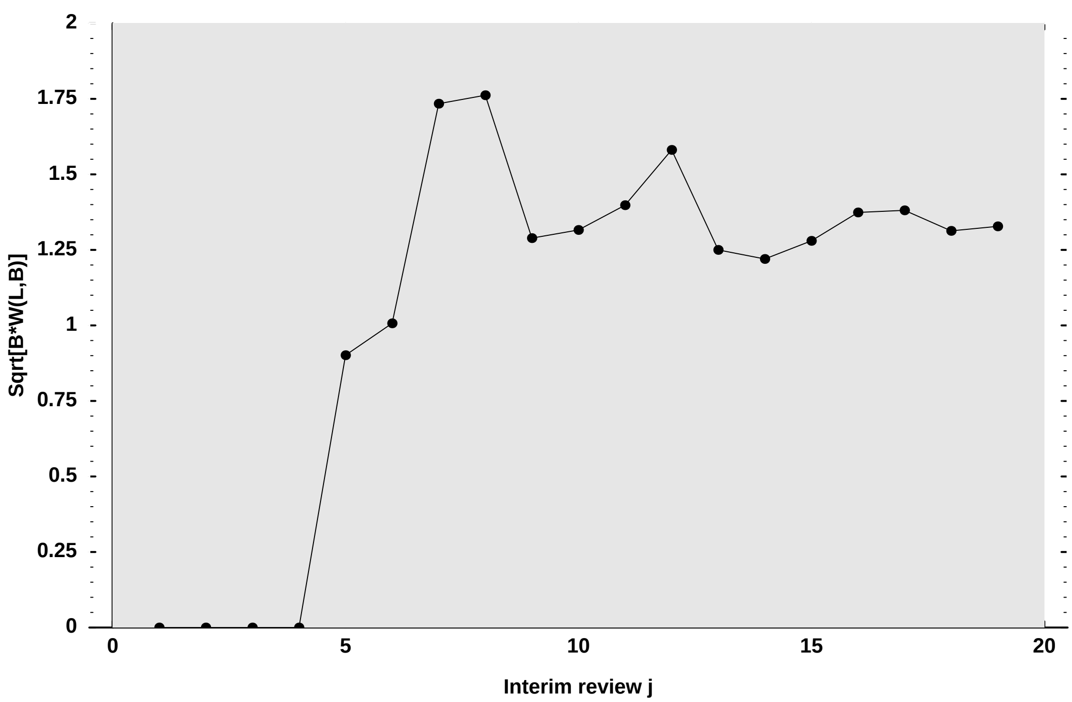


Figure 3: LABATCH.2 estimates, $\sqrt{BW(L,B)}$, of σ_∞

(a) Series 1, $\sigma_\infty = 189.5$



(b) Series 2



and for Series 2 for estimates computed on review $j \geq 7$.

For any path length, $t \geq 20$, LABATCH.2 automatically computes the number of batches, L, and the batch size, B, to be used in its first review. For example, it chose L=7 and B=5 for $t = 10^7$ for the M/M/1 example. This automatic selection relieves the user of the need to specify initial values for L and B, as in the original LABATCH package. Section 2.4 describes how LABATCH.2 makes this selection.

A call from a C / FORTRAN / SIMSCRIPT II.5 main program to

BATCH_MEANS(IN_UNIT,OUT_UNIT,T,S_NUM,PSI_VECTOR, DELTA,RULE, BETA,L_UPPER,SCREEEN) (1)

implements LABATCH.2, where Table 1 defines its arguments. As an example, suppose they assume the values:

IN_UNIT	=	0	
OUT_UNIT	=	15	
T	=	1000	
S_NUM	=	2	
PSI_VECTOR	=	pointer to data vector	
DELTA	=	.01	(2)
RULE	=	1	
BETA	=	.10	
L_UPPER	=	30	
SCREEN	=	0 .	

Then LABATCH.2 processes S_NUM=2 sequences in T=1000 iterative calls from the user's main program and write the output to a file called c.15 / fort.15 / SIMU15 . Moreover, it employs the ABATCH rule to determine batch size on each review and tests for independence of batch averages at the BETA=.10 signifiance level. LABATCH.2 begins its first iteration with the number of batches no greater than L_UPPER = 30.

For RULE=1 (ABATCH), testing determines whether the batch size on review $j+1$ increases by a factor of 2 (rejection) or approximately as $\sqrt{2}$ (success). A doubling of batch size aims at reducing any residual systematic error detected on review j as fast as possible. A $\sqrt{2}$ increase signals that the number of batches are also increasing approximately by a $\sqrt{2}$ factor. This growth in both batch size and number batches as the sample path length grows is a necessary condition for obtaining a consistent estimator of σ_∞^2. Recall that σ_∞^2/t approximates the large-sample variance of $\bar{X}_t$.

2 Rationale for LABATCH.2

Let $\{X_i,\ i \geq 1\}$ denoted a strictly stationary stochastic process with unknown mean μ. Given a sample path, $X_1, \ldots, X_t$,

Table 1: LABATCH.2 arguments

User-specified input	**Definition**	**Notation in Paper**
IN_UNIT	:= 0 if data are to be repeatedly transferred from a main program	...
	:= 30 if data are to be read one vector at a time from input file c.15 / fort.15 / SIMU15 n.b. 30 is merely an example	
OUT_UNIT	:= designated unit for writing output	...
T	:= total number of observations	t
S_NUM	:= number of sample series	... (3)
PSI_VECTOR	:= pointer to vector of S_ NUM sample values	...
DELTA	:= desired significance level for confidence intervals (.01 or .05 suggested)	δ
RULE	:= 1 if ABATCH rule is used := 0 if LBATCH rule is used	
BETA	:= significance level for testing for independence (.10 suggested) n.b. If BETA = 1, each review uses the FNB rule = -1, each review uses the SQRT rule	β
L_UPPER	:= upper bound on initial number of batches (3 <= L_UPPER <= 100)	...
SCREEN	:= 1 if interim review estimates are to be displayed on the screen := 0 otherwise	

$$\bar{X}_t := \frac{1}{t}\sum_{i=1}^{t} X_i$$

provides a point estimate of μ. When data collection begins in the steady state, $\bar{X}_t$ is an unbiased estimator of μ.

Assumption 1. $\sigma_t^2 := \text{var}\bar{X}_t$ satisfies $t\sigma_t^2 \to \sigma_\infty^2 \quad$ as $t \to \infty$, where σ_∞^2 is a positive constant.

Assumption 2. There exist a constant $\lambda \in (0, 1/2)$ such that

$$t^{1/2}(\bar{X}_t - \mu)/\sigma_\infty = Z(t)/t^{1/2} + \mathrm{O}(t^{-\lambda}) \text{ as } t \to \infty \text{ w.p.1},$$

where $\{Z(s), s \geq a\}$ denotes standard Brownian motion.

Assumption 2 is the *Assumption of Strong Approximation* (ASA). A λ close to 1/2 signifies a marginal distribution for the X_i close to the standard normal and low correlation between X_i and X_j for $\forall\, i \neq j$. Conversely, λ close to zero implies the absence of one or both of these properties. See Philipp and Stout (1975). Section 2.4 relies on assumptions 1 and 2.

To assess how well $\bar{X}_t$ approximates μ, we need an estimate of σ_∞^2. The batch means method offers one option. Let $b(t)$ denote a positive integer ($< t$), let $l(t) := \lfloor t/b(t) \rfloor$, and let $t'(t) := l(t)b(t)$. Our version of the batch means method partitions the sequence, $X_1, \ldots, X_{t'(t)}$, into $l(t)$ nonoverlapping batches each of size $b(t)$, computes the batch averages,

$$Y_{jb(t)} := \frac{1}{b(t)} \sum_{i=1}^{b(t)} X_{(j-1)b(t)+i} \quad j = 1, \ldots, l(t), \tag{4}$$

and an estimate of $\text{var}Y_{jb(t)}$,

$$W_{l(t)b(t)} := \frac{1}{l(t)-1} \sum_{j=1}^{l(t)} (Y_{jb(t)} - \bar{X}_{t'(t)})^2, \tag{5}$$

and uses $b(t)W_{l(t)b(t)}$ as an estimte of σ_∞^2. Then

$$[\bar{X}_t \pm \tau_{l(t)-1}(1-\delta/2)\sqrt{b(t)W_{l(t)b(t)}/t}\,] \tag{6}$$

provides an approximating $100 \times (1-\delta)$ percent confidence interval for μ, where $\delta \in (0,1)$ and $\tau_{l(t)-1}(1-\delta/2)$ denotes the $1-\delta/2$ critical value of Student's t distribution with $l(t)-1$ degrees of freedom.

If $\{b(t)\}$ and $\{l(t)\}$ are nondecreasing in t, $b(t) \to \infty$ and $l(t) \to \infty$ as $t \to \infty$, $t^{1/2}\lambda(\log t)/b(t) \to 0$ as $t \to \infty$, and there exists a positive integer $q < \infty$ such that

$$\sum_{i=1}^{\infty} l^{-q}(i) < \infty,$$

then $b(t)W_{l(t)b(t)}$ is a strongly consistent estimator of σ^2_∞ and $(\bar{X}_t - \mu)/\sqrt{b(t)W_{l(t)b(t)/t}} \xrightarrow{d} \mathscr{N}(0,1)$ as $t \to \infty$ (Damerdji 1994). These conditions merely place bounds on the rate at which batch size can grow with sample path length and imply that expression (6) is an asymptotically valid confidence interval for μ in the sense that it achieves the specified $1-\delta$ coverage rate as $t \to \infty$.

Unless clarity demands othewise, we hereafter write $b := b(t)$ and $l := l(t)$ when batch size and number of batches are deterministic function of t. Also, we assume that $t'(t) = t$ so that the batches in expression (4) use all the observations. Later, Secs. 2.2 through 2.5, consider the more general case of $t'(t) \le t$.

Expression (6) provides an assessment of how well $\bar{X}_t$ approximates μ. However, since t is always finite in practice, it implies a distributional approximation that affects the true coverage rate relative to the specified theoretical coverage rate, $1-\delta$. Let

$$Z_{lb} := \frac{\bar{X}_t - \mu}{\sqrt{W_{lb}/l}}. \tag{7}$$

and let F_{lb} denote the distribution function (d.f.) of Z_{lb} with Edgeworth expansion (Chien 1989)

$$F_{lb}(z) = \Phi(z) + \left\{ \frac{\kappa_3(Z_{lb})}{6} - \kappa_1(Z_{lb}) + \left[\frac{\kappa_4(Z_{lb})}{8} - \frac{\kappa_2(Z_{lb})-1}{2} \right] z - \frac{\kappa_3(Z_{lb})}{6} z^2 \right.$$
$$\left. - \frac{\kappa_4(Z_{lb})}{24} z^3 + \mathrm{O}(1/l) \right\} \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-z^2/2} \qquad -\infty < z < \infty,$$

where Φ denotes the standard normal d.f. and $\kappa_i(Z_{lb})$ denotes the ith cumulant of Z_{lb}. If $\mathrm{E}|X_1 - \mu|^{20} < \infty$ and $\{X_i\}$ is ϕ-mixing with $\phi_i(i^{-13})$, then (Chien 1989, p.46)

$$\begin{aligned}
\kappa_1(Z_{lb}) &= \mathrm{O}(1/l^{1/2})\mathrm{O}(1/b^{1/2}) \\
\kappa_2(Z_{lb}) &= 1 + \mathrm{O}(1/l) + \mathrm{O}(1/b) \\
\kappa_3(Z_{lb}) &= \mathrm{O}(1/l^{1/2})\mathrm{O}(1/b^{1/2}) \\
\kappa_4(Z_{lb}) &= \mathrm{O}(1/l).
\end{aligned}$$

Actually, Chien gives slightly weaker conditions for the first three cumulants.

As an immediate consequence for fixed z,

$$F_{lb}(z) - \Phi(z) = \mathrm{O}(1/l^{1/2})\mathrm{O}(1/b^{1/2}) + \mathrm{O}(1/l) + \mathrm{O}(1/b), \tag{8}$$

revealing that choosing $l(t) \propto t^{1/2}$ and $b(t) \propto t^{1/2}$ induces the fastest convergence of the true coverage rate to the specified theoretical coverage rate $1-\delta$.

When $X_1, \ldots, X_t$ are i.i.d., the assignment $b(t) = 1$ and $l(t) = t$ makes expression (6) an approximating confidence interval based on $t-1$ degrees of freedom whose error of approximation is generally negligible, say, for $t \ge 100$, regardless of the parent distribution of the X_i. When $\{X_i\}$ is a dependent sequence, no comparable rule-of-thumb applies since two properties now affect the error of approximation. As before, one is the marginal distribution of the X_i; the other is the extent and character of the correlation between each X_i and X_j for $\forall i \neq j$ in the sample. An analysis of the properties of $V_t := b(t)W_{l(t)b(t)}$ reveals how these errors arise.

Observe that for $t'(t) = t$

$$\mathrm{E}V_t = \frac{t}{l-1}(\sigma_b^2 - \sigma_t^2), \tag{9}$$

which, for positively autocorrelated sequences, is usually negative. If $\mathrm{E}(X_1 - \mu)^{12} < \infty$ and $\{X_i\}$ is ϕ-mixing with $\phi_i = \mathrm{O}(i^{-9})$, then expression (9) takes the form (Goldsman and Meketon 1986)

$$\mathrm{E}\ V_t = \sigma_\infty^2 + \gamma(l+1)/t + \mathrm{o}(1/b), \tag{10}$$

where

$$\gamma := -2\sum_{i=1}^{\infty} i\,\mathrm{cov}(X_1, X_{1+i}).$$

Also (Chien et al. 1996),

$$\mathrm{var}\ V_t = \frac{2\sigma_\infty^4(l+1)}{(l-1)^2} + \mathrm{O}(1/lb^{1/4}) + \mathrm{O}(1/l^2). \tag{11}$$

Note that $\sigma_\infty^2 = \sum_{i=\infty}^{\infty} \mathrm{cov}(X_1, X_{1+i})$.

Expression (9) leads to the representation

$$V_t - \sigma_\infty^2 = \underbrace{t\sigma_t^2 - \sigma_\infty^2}_{\substack{\text{error due} \\ \text{to finite } t}} \quad \underbrace{- t\sigma_t^2\left(\frac{1 - b\sigma_b^2/t\sigma_t^2}{1 - b/t}\right)}_{\substack{\text{error due to} \\ \text{ignoring correlation} \\ \text{between batches}}} \quad + \quad \underbrace{\epsilon_t,}_{\substack{\text{error due to} \\ \text{random sampling}}} \tag{12}$$

where ϵ_t has mean zero and variance (11). Hereafter, we collectively refer to the errors due to finite t and to ignoring correlation as systematic error. From expression (10) we see that the systematic error behaves as $\mathrm{O}((l+1)/t) = \mathrm{O}(1/b)$ whereas, from expression (11), $\sqrt{\mathrm{var}\ \epsilon_t}$ behaves as $\mathrm{O}(1/l^{1/2})$, revealing the tradeoff between the two types of error that a choice of l and b induces.

For $l \propto t^{1/2}$ and $b \propto t^{1/2}$, systematic error in V_t and random error, ϵ_t, both decrease as $\mathrm{O}(1/t^{1/2})$, whereas for each z the absolute deviation in coverage rates, $|F_{lb}(z) - \Phi(z)|$ also diminishes as $\mathrm{O}(1/t^{1/2})$. However, representations (10) and (11) imply that $l \propto t^{2/3}$ and $b \propto t^{1/3}$ induce the fastest convergence, $\mathrm{O}(1/t^{2/3})$, for the mean-square error of V_t (Goldsman and Meketon 1986, Chien et al. 1996, and Carlstein 1985 for a first-order-autoregressive model with normal residuals). But this choice induces the slower rate, $\mathrm{O}(1/t^{1/3})$, for $|F_{lb}(z) - \Phi(z)|$.

In what follows, we adopt a strategy that leads to $l \propto t^{1/2}$ and $b \propto t^{1/2}$, once t becomes sufficiently large. Two considerations motivate this choice. First, since the approximating

$100\times(1-\delta)$ percent confidence interval (6) provides the means of assessing how well $\bar{X}_t$ approximates μ, we would like its true coverage rate to be as close as possible to $1-\delta$ as possible. Secondly, the square root option facilitates the efficient computation of a sequence of estimates $V_{t_1}, V_{t_2}, \ldots,$ based on subsequences $\{X_1, \ldots, X_{t_1}\}$, $\{X_1, \ldots X_{t_2}\}, \ldots,$ respectively, for $t_1 < t_2 < \ldots \leq t$, that provide a means for assessing the extent to which systematic error remains in V_t, our final estimate of σ^2_∞. The next subsection elaborates on this issue.

2.1 Interim Review

If an analyst computes only one estimate, $W_{l(t)b(t)}$, based on the total sample record of length t, then, regardless of the batch size assignment rule, the analysis provides no information about the extent to which the desired asymptotic properties hold for V_t as an approximation to σ^2_∞. A sequence of *interim reviews* overcomes this limitation by computing and displaying results for overlapping segments of the sample record of successively increasing lengths $t_1 < t_2 < \ldots \leq t$. The approach has considerable value when the cost per observation is high and there is doubt, before experimentation, about how many observations to collect. Moreover, a judiciously chosen growth factor for review times allows LABATCH.2 to compute the sequence of estimates $W_{l(t_1)b(t_1)}$, $W_{l(t_2)b(t_2)}, \ldots, W_{l(t_j)b(t_j)}$, $W_{l(t)b(t)}$ in $\mathrm{O}(t)$ time, employing $\mathrm{O}(\log_2 t)$ space. Maintaining an $\mathrm{O}(t)$ computing time is essential for the single sample path approach to achieve greater statistical efficiency than the multiple independent replications approach. Moreover, if generating the data takes $\mathrm{O}(t)$ time, then no increase in computational time complexity arises. The $\mathrm{O}(\log_2 t)$ space complexity is particularly appealing when t is large and interim reviews are desired for estimates of more than one mean.

In what follows we take t_1 as given and $t_{j+1} := 2t_j$ for $j = 1, 2, \ldots$. For a sequence of i.i.d. random variables $X_1, X_2, \ldots,$ this choice implies $\mathrm{corr}(\bar{X}_{t_j} \bar{X}_{t_{i+j}}) = \sqrt{t_j/t_{i+j}}$ for $i \geq 0$ and $j \geq 1$. More generally, if $\mathrm{corr}(X_1, X_{i+j}) = \alpha^{|j|}$, for some $-1 < \alpha < 1$, then $\lim_{j\to\infty} \mathrm{corr}(\bar{X}_{t_j}, \bar{X}_{t_{i+j}}) = \lim_{j\to\infty} \sqrt{t_j/t_{i+j}}$. These observations suggest that we choose $\theta := t_{j+1}/t_j$ sufficiently large so that estimates on success reviews are not substantially redundant. We choose $\theta = 2$, which implies $\mathrm{corr}(\bar{X}_{t_j}, \bar{X}_{t_{i+j}}) = 2^{-i/2}$ in the independent and asymptotic (as $j \to \infty$) cases. For example, $\mathrm{corr}(\bar{X}_{t_j}, \bar{X}_{t_{j+1}}) = \sqrt{2}/2 = .7071$ (in the i.i.d. case). Any growth factor less than 2 would induce a larger correlation. Also, this choice makes possible the $\mathrm{O}(t)$ computing time and the $\mathrm{O}(\log_2 t)$ space bounds for LABATCH.2.

2.2 FNB and SQRT Rules

The description here closely follows Fishman (1996) and Fishman and Yarberry (1997). For $j = 1, 2, \ldots,$ let $l_j := l(t_j)$ and $b_j := b(t_j)$. We illustrate the benefits of interim review for two assignment rules that eventually (Sec. 2.3) form the basis for the LBATCH and ABATCH rules that LABATCH.2 incorporates. Given (l_1, b_1), the FNB rule fixes $l_j = l_1$ for all j and doubles the batch size $b_{j+1} = 2b_j$ on successive reviews. Procedure FNB describes its principal steps.

Procedure FNB (LABATCH.2)

Given: l_1, b_1, and $t(> l_1 b_1)$
$l \leftarrow l_1$ (fix # of batches)
$J(t) \leftarrow 1 + \lfloor \log(t/l_1 b_1)/\log 2 \rfloor$ (# of reviews)
$t'(t) \leftarrow 2^{J(t)-1} l_1 b_1$ (# of observations used on review $J(t)$)
$j \leftarrow 1, t_0 \leftarrow 0, t_1 \leftarrow l_1 b_1,$
While $t_j \leq t'(t)$:
  Collect $X_{t_{j-1}+1}, \ldots, X_{t_j}$ additional observations
  Compute $\bar{X}_{t_j}$
  Form batches of size b_j based on $X_1, \ldots, X_{t_j}$
  Compute W_{lb_j}
  $b_{j+1} \leftarrow 2b_j$
  $j \leftarrow j+1$
  $t_j \leftarrow 2t_{j-1} \quad (= l_1 b_j)$
Collect observations $X_{t'(t)+1}, \ldots, X_t$
Compute $\bar{X}_t$
Output:
  $\bar{X}_t$ is the point estimate of μ.
  $[\bar{X}_t \pm \tau_l(1-\delta/2)\sqrt{b_{J(t)} W_{lb_{J(t)}}/t}\ ]$ is the approximating
    $100 \times (1-\delta)$ percent confidence interval for μ.
  $\{b_j W_{lb_j};\ j = 1, \ldots, J(t)\}$ is the sequence of successive estimates of σ^2_∞.

Given (l_1, b_1), the SQRT rule sets

$$l_2 = \tilde{l}_1 := \lfloor \sqrt{2} l_1 + .5 \rfloor \tag{13}$$

$$b_2 = \tilde{b}_1 := \begin{cases} 3 & \text{if } b_1 = 1 \\ \lfloor \sqrt{2} b_1 + .5 \rfloor & \text{if } b_1 > 1 \end{cases} \tag{14}$$

$$\begin{aligned} l_{j+1} &= 2 l_{j-1} \\ b_{j+1} &= 2 b_{j-1} \quad j = 2, 3, \ldots \ . \end{aligned}$$

Procedure SQRT describes its principal steps.

Procedure SQRT (LABATCH.2)

Given: l_1, b_1, and $t(> l_1 b_1)$
$\tilde{l}_1 \leftarrow \lfloor \sqrt{2}\ l_1 + .5 \rfloor$
$\tilde{b}_1 \leftarrow 3$; if $b_1 > 1, \tilde{b}_1 \leftarrow \lfloor \sqrt{2}\ b_1 + .5 \rfloor$

$J(t) \leftarrow 1 + \lfloor \log(t/l_1 b_1)/\log 2 \rfloor$ (# of reviews)
$t'(t) \leftarrow 2^{J(t)-1} l_1 b_1$ (# of observations used on review $J(t)$)
$j \leftarrow 1,\ t_0 \leftarrow 0,\ t_1 \leftarrow l_1 b_1,\ t_2 \leftarrow \tilde{l}_1, \tilde{b}_1$
While $t_j \leq t'(t)$:
  Collect $X_{t_{j-1}+1}, \ldots, X_{t_j}$ additional observations
  Compute $\bar{X}_{t_j}$
  Compute $W_{l_j b_j}$
  If $j = 1$:
    $l_{j+1} \leftarrow \tilde{l}_1$ and $b_{j+1} \leftarrow \tilde{b}_1$
  Otherwise:
    $l_{j+1} \leftarrow l_{j-1}$ and $b_{j+1} \leftarrow b_{j-1}$
  $j \leftarrow j + 1$
  $t_j \leftarrow l_j b_j$
Collect observations $X_{t'(t)+1}, \ldots, X_t$
Compute $\bar{X}_t$
Output:
  $\bar{X}_t$ is the point estimate of μ.
  $[\bar{X}_t \pm \tau_{l_{J(t)}-1}(1 - \delta/2)\sqrt{b_{J(t)} W_{l_{J(t)} b_{J(t)}}\ /t}\ ]$ is the approximating
    $100 \times (1 - \delta)$ percent confidence interval for μ.
  $\{b_j W_{l_j b_j};\ j = 1, \ldots, J(t)\}$ is the sequence of successive estimates of σ^2_∞.

These assignments induce $l_{j+1}/l_j \doteq \sqrt{2}$ and $b_{j+1}/b_j \doteq \sqrt{2}$. By choosing (l_1, b_1) from $\mathscr{B}$ in Table 2, we ensure that $2 l_1 b_1 = \tilde{l}_1 \tilde{b}_1$ so that $t_j = l_j b_j = 2^{j-1} l_1 b_1$ and, therefore, $t_{j+1}/t_j = 2$, as desired. This constraint proves valuable in Sec. 2.3, which describes batch-size rules that combine the FNB and the SQRT rules.

As illustration, we again use the M/M/1 simulation example. Consider a steady-state sample record, $X_1, \ldots, X_t$, of waiting times in queue for which $\mu = 9$ and $\sigma^2_\infty = 35910$ (Blomqvist 1967). Also, $t = 10^7, l_1 = 7$, and $b_1 = 5$, so that $\tilde{l}_1 = 10$ and $\tilde{b}_1 = 7$, and thus $2 l_1 b_1 = \tilde{l}_1 \tilde{b}_1$, $J(t) = 19$, and $t'(t) = 9{,}175{,}040$. These assignments cause $B_{19} W_{l_{19} b_{19}}$, the final estimate of σ^2_∞ for Series 1 in Fig. 1b, to use 91.75 percent of the data, $X_1, \ldots, X_{9{,}175{,}040}$. Section 2.4 returns to this issue of data usage.

Figure 4 shows $\{\sqrt{b_j W_{l_j b_j}};\quad j = 1, \ldots, 19\}$ for the FNB and SQRT rules. After review 12, the FNB sequence fluctuates around the true σ_∞. The SQRT sequence comes close to σ_∞ after review 15. This contrast in behavior reflects the $O(1/t_j)$ systematic error dissipation rate in expression (11) for the FNB rule as compared to the $O(1/t_j^{1/2})$ rate for the SQRT rule. However, sampling fluctuations for the FNB sequence are $O(1)$, in contrast to $O(1/t_j^{1/2})$ for the SQRT rule.

2.3 LBATCH and ABATCH Rules

Let H denote the hypothesis: On review j, the l_j batches $Y_{1 b_j}, \ldots, Y_{l_j b_j}$ are mutually independent. The LBATCH and ABATCH rules both use the outcome of a test of H to switch

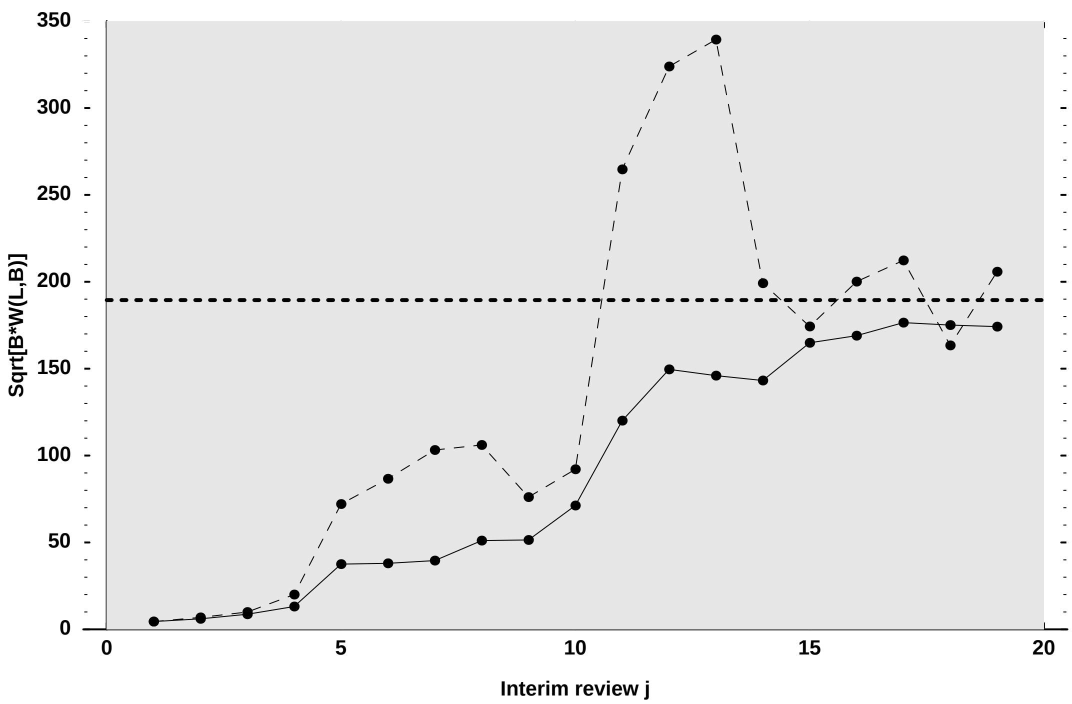
Table 2: $\mathscr{B} := \{(l_1, b_1) : 1 \leq b_1 \leq l_1 \leq 100,\ 2l_1b_1 \ = \ \tilde{l}_1\tilde{b}_1\}^{\dagger}$

b_1	l_1	b_1	l_1
$\underline{1}$	$\underline{3}$	38	54
2	3,6	$\underline{39}$	$\underline{55}$
3	4	$\underline{41}$	58,$\underline{87}$
4	6	$\underline{43}$	$\underline{61}$
$\underline{5}$	$\underline{7}$,14,$\underline{21}$,28	44	62, 93
$\underline{7}$	10,$\underline{15}$,20,$\underline{25}$,30,$\underline{35}$	$\underline{46}$	$\underline{65}$
$\underline{8}$	$\underline{11}$	48	51,68,85
$\underline{9}$	$\underline{13}$	$\underline{49}$	$\underline{69}$
10	14, 21,28	$\underline{50}$	$\underline{71}$
$\underline{12}$	$\underline{17}$,34,$\underline{51}$,68,$\underline{85}$	$\underline{51}$	$\underline{60}$,72,$\underline{84}$,96
14	15,20,25,30,35	$\underline{53}$	$\underline{75}$
$\underline{15}$	$\underline{21}$,28	54	57
$\underline{17}$	24,$\underline{36}$,48,$\underline{60}$,72,$\underline{84}$,96	55	78
$\underline{19}$	$\underline{27}$,54	$\underline{56}$	$\underline{79}$
20	21, 28	58	82
$\underline{21}$	$\underline{25}$, 30, $\underline{35}$	$\underline{60}$	68,$\underline{85}$
$\underline{22}$	$\underline{31}$,62,$\underline{93}$	61	86
24	34,51,68,85	62	66
25	28	$\underline{63}$	$\underline{89}$
$\underline{26}$	$\underline{37}$	65	92
$\underline{27}$	38,$\underline{57}$	$\underline{66}$	$\underline{93}$
28	30,35	$\underline{67}$	$\underline{95}$
$\underline{29}$	$\underline{41}$,82	68	72,84,96
$\underline{31}$	44,$\underline{66}$,88	$\underline{70}$	$\underline{99}$
32	45	72	85
$\underline{33}$	$\underline{47}$	82	87
34	36,48,60,72,84,96	$\underline{84}$	$\underline{85}$
$\underline{36}$	$\underline{51}$,68,$\underline{85}$	$\underline{85}$	$\underline{96}$
37	52	88	93

$^{\dagger}\tilde{l}_1$ is defined in expression (13) and $\tilde{b}_1$ in expression (14).

between the FNB and SQRT rules on succesive reviews. The net effect is to retain the desirable properties of each rule while reducing the influence of their limitations. The principal features of these hybrid rules are:

Figure 4: LABATCH.2 estimation of σ_∞ for waiting time in queue
(FNB rule: light dashed line, SQRT rule: solid line, $\sigma_\infty = 189.3$: heavy dashed line)



LBATCH Rule

- Start with the FNB rule on review 1.
- For $j \geq 1$, if H is rejected on review j, use the FNB rule on review $j+1$.
- Once H is accepted on review j, use the SQRT rule on reviews, $j+1, j+2, \ldots$.

and:

ABATCH Rule

- Start with the FNB rule on review 1.
- For $j \geq 1$, if H is rejected on review j, use the FNB rule on review $j+1$.

- For $j \geq 1$, if H is accepted on review j, use the SQRT rule on review $j+1$.

By initially fixing l, the LBATCH rule allows batch size, b, to increase at the maximal rate when H is rejected, thus dissipating systematic error in V_{t_j} as fast as possible. Once H is accepted, the rule switches to the SQRT rule to dissipate the error in coverage rate as rapidly as possible. By testing H on every review, the ABATCH rule takes into consideration the possibility of Type II errors on successive replications. The LBATCH rule ignores this source of error whereas the ABATCH rule allows the procedure to correct itself.

As a consequence of this testing, batch size and number of batches on review j (> 1) are random. To acknowledge this property, we denote them as B_j and L_j respectively. Let K_j denote the number of test rejections of H on reviews $1, \ldots, j$. Then the LBATCH and ABATCH rules induce:

$$L_j = \begin{cases} 2^{(j-K_j)/2} l_1 & \text{if } j - K_j \text{ is even} \\ 2^{(j-K_j-1)/2} \tilde{l}_1 & \text{if } j - K_j \text{ is odd} \end{cases}$$

and

$$B_j = \begin{cases} 2^{(j+K_j-2)/2} b_1 & \text{if } j - K_j \text{ is even} \\ 2^{(j+K_j-1)/2} \tilde{b}_1 & \text{if } j - K_j \text{ is odd,} \end{cases}$$

so that $t_j = L_j B_j = 2^{j-1} l_1 b_1$ for $j = 1, 2, \ldots$.

To account for the randomness of $\{(L_j, B_j),\ j \geq 1\}$ with regard to limiting behavior, we rely on:

Theorem 1. (Fishman and Yarberry 1997). If Assumptions 1 and 2 hold, $2 l_1 b_1 = \tilde{l}_1 \tilde{b}_1$, and there exist constants $\Gamma < 1$ and $\theta > 1 - 4\lambda$ such that $\sum_{j=1}^{\infty} \text{pr}\ (K_j/j > \Gamma) < \infty$ and $K_j/j \to \theta$ as $j \to \infty$ w.p.1, then

$$B_j W_{L_j B_j} \to \sigma_\infty^2 \quad \text{as } j \to \infty \quad \text{w.p.1}$$

and

$$\frac{\bar{Y}_{t_j} - \mu}{\sqrt{W_{L_j B_j}/L_j}} \xrightarrow{\text{d}} \mathcal{N}(0,1) \quad \text{as } j \to \infty.$$

□

Provided that its conditions are met, the theorem gives a consistent estimator of σ_∞^2 and the basis for the asymptotic validity of the interval (6), as Sec. 2.4 shows. We conjecture that there is a large class of problems that satisfy these conditions.

2.4 Choosing t, l_1, and b_1

For given $(l_1, b_1) \in \mathscr{B}$, choosing t so that $\log(t/l_1 b_1)/\log 2$ is an integer results in $t'(t) = t$. As a consequence, the LBATCH and ABATCH rules used all t observation to estimate $W_{L_{J(t)}B_{J(t)}}$ and $\bar{Y}_{t'(t)} = \bar{X}_t$. Since choosing l_1, b_1, and t subject to the constraint may be too burdensome for some users, LABATCH.2 merely requires that t be specified and then chooses l_1 and b_1 to maximize $t'(t)$.

Let $\mathscr{B}(t)$ denote the subset of $\mathscr{B}$ that maximizes $t'(t)$. LABATCH.2 chooses (l_1, b_1) to be the element of $\mathscr{B}(t)$ that maximizes $J(t)$, the number of interim reviews. As the following example shows, this algorithm reduces the number of elements in $\mathscr{B}$ that need to be considered when maximizing $t'(t)$. Suppose that (a_1, a_2) and $(a_3, a_4) \in \mathscr{B}$ maximize $t'(t)$ and that for some integer $\alpha \geq 1$ $a_1 a_2 = 2^\alpha a_3 a_4$. Then clearly the choice (a_1, a_2) induces more reviews than (a_3, a_4). As a consequence of this example, LABATCH.2 only considers the underlined 2-tuples in $\mathscr{B}$ in Table 2 when maximizing $t'(t)$. As illustration, $t'(10^7) = 9,175,040$ is maximal for each 2-tuple in

$$\mathscr{B}(10^7) = \{(7,5),(14,5),(25,5),(10,7),(20,7),(14,10),(28,10),(20,14),(28,20)\},$$

but (7,5) allows for the maximal number of reviews, $J(10^7) = 19$.

For $10 \leq t \leq 10^7$, Table 3 shows the worst-case proportion of observations used for $W_{L_{J(t)}B_{J(t)}}$ when $t'(t)$ is maximized subject to alternative upper bounds on l_1. Recall that $l_1 \leq 100$ implies optimization over all entries in $\mathscr{B}$. In this case, choosing $t \geq 500$ implies that no less than 89.8 percent of the data is used for computing $W_{L_{j(t)}B_{J(t)}}$. Alternatively, imposing the constraint, $l_1 \leq 30$, when choosing (l_1, b_1) to maximize $'(t)$, reduces this percentage to 88.9. We recommend either of these options for $t \geq 500$.

Table 3: Data utilization for final estimate of σ^2_∞

$$\min_t \left[\frac{1}{t} \max_{(l_1,b_1)\in\mathscr{B}} t'(t)\right]$$

	$10 \leq t \leq 24$	$25 \leq t \leq 49$	$50 \leq t \leq 99$	$100 \leq t \leq 499$	$500 \leq t \leq 10^7$
$l_1 \leq 10$	.522	.706	.696	.688	.686
$l_1 \leq 20$	.522	.706	.696	.798	.795
$l_1 \leq 30$	.522	.706	.696	.805	.889
$l_1 \leq 100$	.522	.706	.696	.805	.898

We now reconcile

$$A_t := \frac{\bar{X}_t - \mu}{\sqrt{B_{J(t)} W_{L_{J(t)}B_{J(t)}}/t}} \tag{15}$$

whose properties establish the basis for the confidence interval in expression (6), with $l(t) = L_{J(t)}, b(t) = B_{J(t)}$, and $W_{l(t)b(t)} = V_{t'(t)}/B_{J(t)}$, and

$$G_t := \frac{\bar{X}_{t'(t)} - \mu}{\sqrt{B_{J(t)} W_{L_{J(t)} B_{J(t)}} / t'(t)}} \tag{16}$$

whose properties are described in Theorem 1 for $j = J(t)$. Recall that $t'(t) = t_{J(t)} = L_{J(t)} B_{J(t)}$. For

$$H_t := \frac{\frac{1}{t - t'(t)} \sum_{i=t'(t)+1}^{t} (X_i - \mu)}{\sqrt{B_{J(t)} W_{L_{J(t)} B_{J(t)}} / [t - t'(t)]}} \ ,$$

we have

$$A_t = \sqrt{\frac{t'(t)}{t}}\ G_t + \sqrt{1 - \frac{t'(t)}{t}}\ H_t.$$

Although $G_t \xrightarrow{\text{d}} \mathcal{N}(0,1)$ as $t \to \infty$ by Theorem 1, the confidence interval (6) is based on A_t, not G_t. To reconcile this disparity we argue as follows:

Since $B_{J(t)} W_{L_{J(t)} B_{J(t)}}$ is a consistent estimator of σ_∞^2, $H_t \xrightarrow{\text{d}} \mathcal{N}(0,1)$ as $t \to \infty$. Moreover,

$$\text{corr}(G_t, H_t) \to 0 \quad \text{as } t \to \infty.$$

Therefore, A_t asymptotically behaves as the sum of two independent normally distributed random variables with mean zero and variance $t'(t)/t + 1 - t'(t)/t = 1$. The use of the critical value from Student's distribution rather than the corresponding value for the standard normal is merely intended to compensate for those cases in which $L_{J(t)}$ turns out to be small.

2.5 Testing H

To test H, we use the von Neumann ratio (von Neumann 1941, Young 1941, Fishman 1978, 1996)

$$C_l := 1 - \frac{\sum_{i=1}^{l-1} (Y_{ib} - Y_{i+1,b})^2}{2 \sum_{i=1}^{l} (Y_{ib} - \bar{X}_t)^2}.$$

If $\frac{Y_{1b}-\mu}{\sigma_b}, \ldots, \frac{Y_{lb}-\mu}{\sigma_b}$ are independent from $\mathcal{N}(0,1)$, then under H, C_l has mean zero, variance $(l-2)/(l^2-1)$ and a distribution that is remarkably close to normal for l as small as eight. For independent non-normal $Y_{1b}, ..., Y_{lb}$ under H, C_l has mean zero. As l increases, $[(l^2 - 1)/(l-2)]$ var $C_l \to 1$ and the *skewness*, $\text{E}(C_l - \text{E}C_l)^3/[\text{E}(C_l - \text{E}C_l)^2]^{3/2}$, and *excess kurtosis*, $\text{E}(C_l - \text{E}C_l)^4/[\text{E}(C_l - \text{E}C_l)^2]^2 - 3$, converge to zero.

If $\{X_i,\, i \geq 0\}$ has a monotone non-increasing autocorrelation function, then a one-sided test of size β is in order. In particular, if $C_l \leq \Phi^{-1}(1-\beta)\sqrt{(l-2)/(l^2-1)}$, H is accepted. Some $\{X_i\}$ have autocorrelation functions that exhibit damped harmonic behavior around the zero axis. In this case, the outcome of the test may be seriously misleading for small batch sizes. Repeated testing on successive reviews under the ABATCH rule reduces the possibility of a misleading outcome, thus giving this rule a distinct advantage over the LBATCH rule.

For purposes of comparison, Fig. 5 displays $\{\sqrt{B_j W_{L_j B_j}};\quad j = 1, \ldots, J(10^7)\}$ for the ABATCH, FNB, and the SQRT rules. The dissipation of systematic error under the ABATCH rule mirrors that under the FNB rule. However, its sampling fluctuations are considerably attenuated in comparison to those for the FNB rule.

Figure 5: LABATCH.2 estimation of σ_∞ for waiting time in queue
(ABATCH rule: heavy line, FNB rule: thin dashed line,
SQRT rule: thin line, $\sigma_\infty = 189.5$: heavy dashed line)

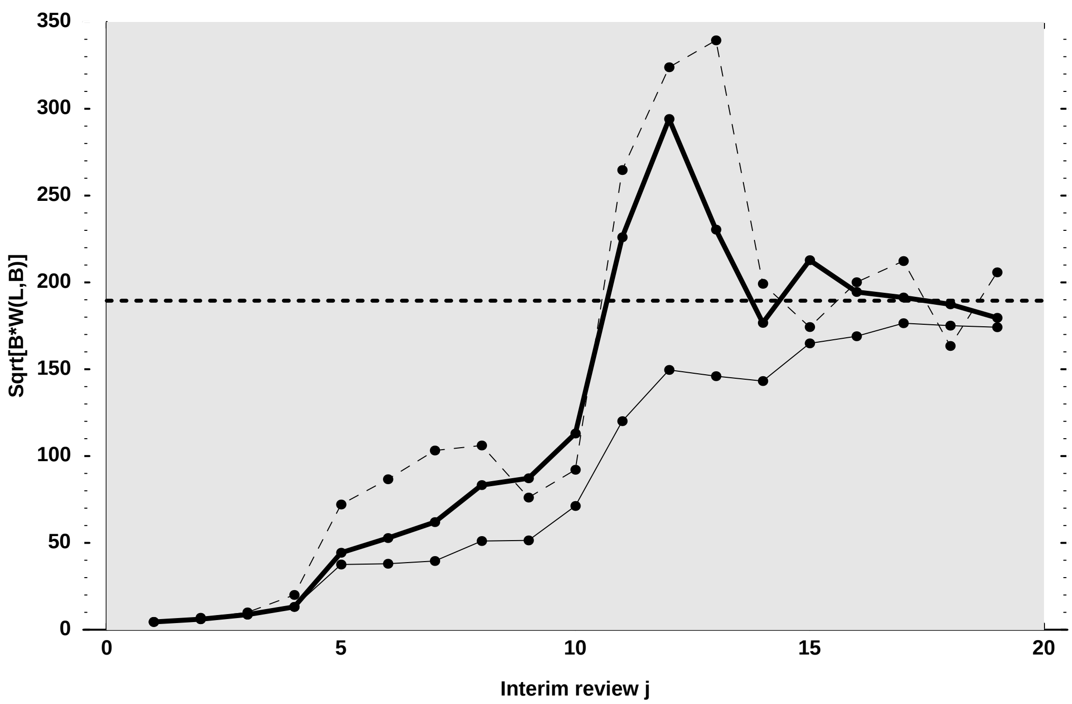


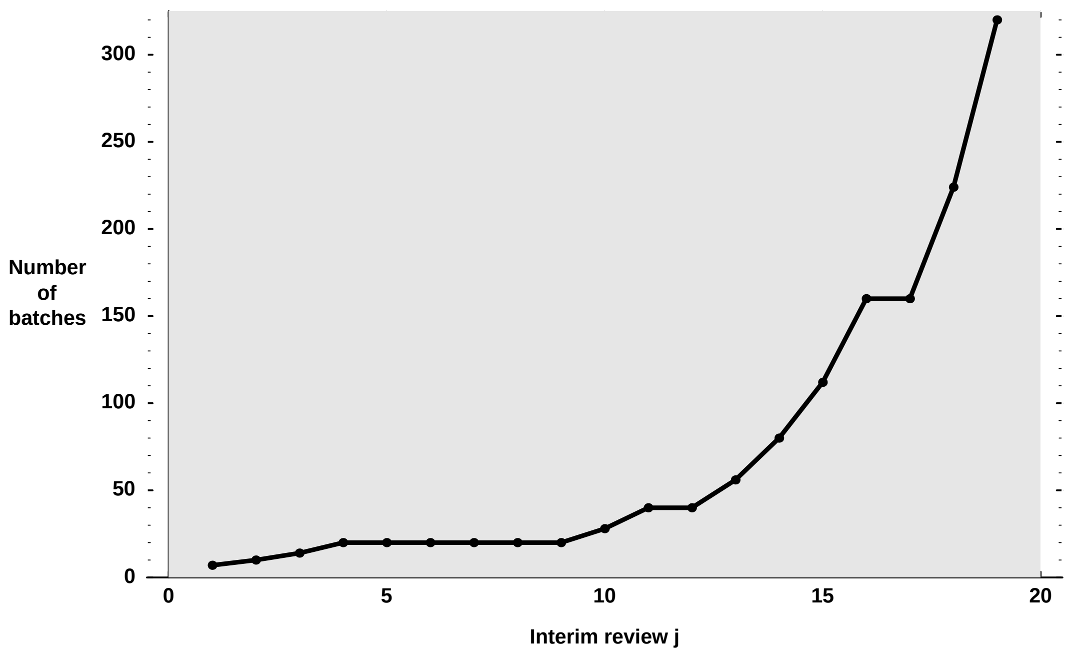
Figure 6a shows $\{L_j;\quad j = 1, \ldots, J(10^7)\}$ and Fig. 6b displays

$$P_j := 1 - \Phi\left(\sqrt{\frac{L_j^2 - 1}{L_j - 2}}\, C_{L_j}\right) \qquad j = 1, \ldots, 19,$$

which we call the p-value. Collectively, they reveal how the sequence $\{B_j W_{L_j B_j}\}$ evolves under the ABATCH rule. If $P_j < \beta = .10$ on review j, then LABATCH.2 rejects H and uses

Figure 6: LABATCH.2 output for
Series 1 using ABATCH rule

(a) Number of batches



(b) p-value for testing H, $\beta = .10$

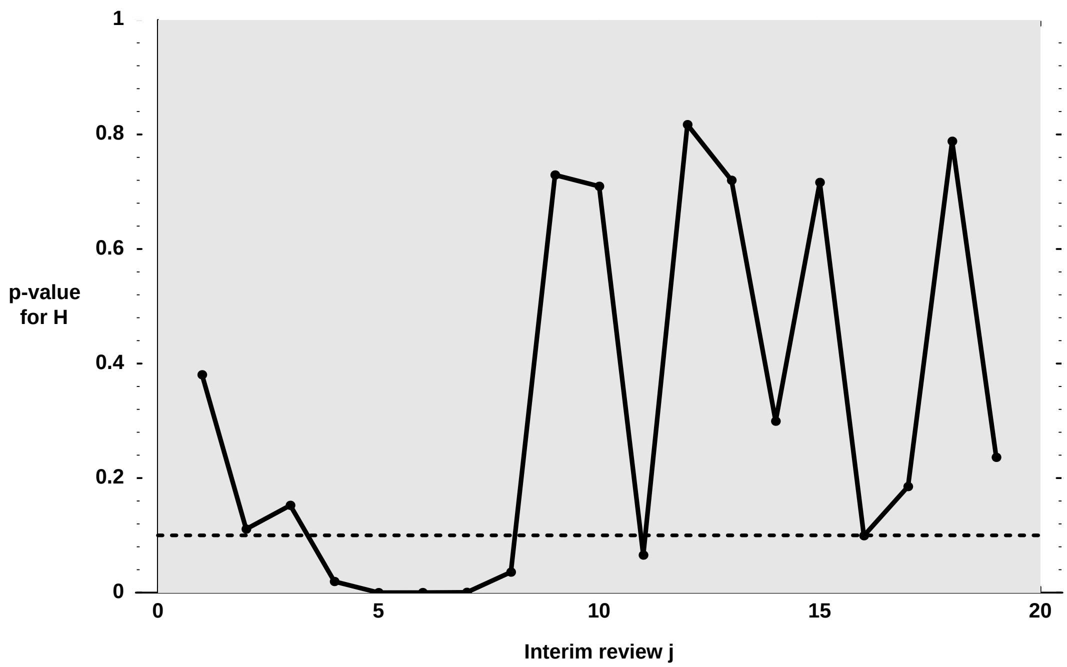
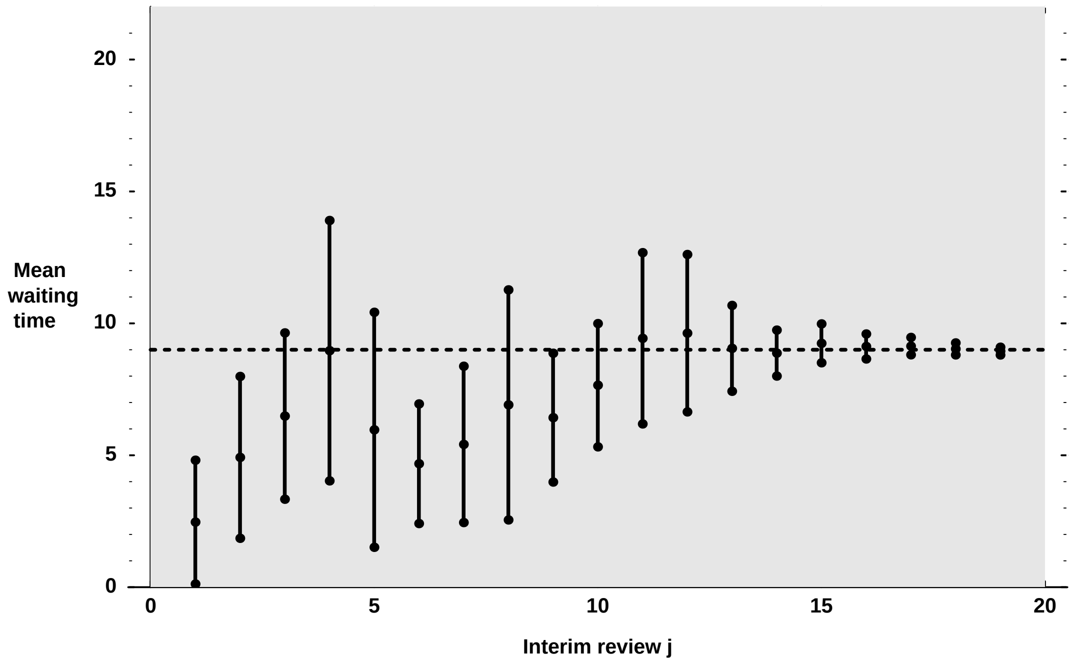


Figure 7: LABATCH.2 Sample means and 99 percent confidence intervals for Series 1 ABATCH rule; simulation starts in empty and idle state



the FNB rule on review $j+1$. This implies $L_{j+1} = L_j$, as Fig. 6a shows on reviews j=1, 5 through 9, 12, and 17. If $P_j \geq \beta = .10$, then LABATCH.2 uses the SQRT rule on review $j+1$ so that $L_{j+1}/L_j \doteq \sqrt{2}$. Figure 6a illustrates this behavior on reviews j=2, 3, 4, 10, 11, 13 through 16, 18, and 19. The long run of successes beginning on review 12 suggests that batch sizes $B_{12} = 10240$ make the residual correlation between batches negligible.

3 Assessing the Adequacy of the Warm-Up Interval

LABATCH.2 also provides an assessment of the extent of warm-up bias in $\bar{X}_t$. We again use the M/M/1 simulation. However, this time the run began with an arrival to an empty and idle system and data collection began with X_1 := waiting time of the first arrival. For this scenario, $X_i = 0$ w.p.1 and $X_1, X_2, \ldots, X_t$ is stationary, only asymptotically (as $t \to \infty$). This is a more biased environment than one would expect to encounter when data collection begins after a user-specified warm-up interval.

Figure 7 displays the 99 percent confidence intervals for mean waiting time in queue, taken from the LABATCH.2 interim review tableau for Series 1 for this run. It suggests little bias after review 11, which corresponds to $2^{11} \times L_1 \times B_1 = 2048 \times 7 \times 5 = 71,680$ observations.

Suppose the user had specified $t = 4480$ for the sample path length. Then LABATCH.2 would have generated the same sample averages and confidence intervals for $L_1 = 7$ and

$B_1 = 5$, but only for the first seven interim reviews. Moreover, a display of these results would have aroused considerably more concern about the dissipation of bias in $\bar{X}_t$. Interestingly, had we taken $t = 4480$ in our first simulation, which began in an equilibrium state, we might equally be suspicious of $\bar{X}_t$ based on the path it displays in Fig. 2 for $j = 1, \ldots, 7$. However, this observation in no way mitigates the value of the assessment when we know for a fact that $\bar{X}_t$ may contain systematic bias as a result of starting in an arbitrarily chosen state and possibly choosing an inadequate warm-up interval.

This example in no way mitigates the traditionally sound advice of truncating a warm-up interval in the sample data to reduce the influence of initials conditions.

4 Features of LABATCH.2

Several features of LABATCH.2 allow wide latitude for using it in practice. For example, one can easily strip the header and trailer entries from the column display in the interim review tableau and then transfer the remaining tableau to a spreadsheet environment, thus facilitates graphmaking. Experience has shown that little effort is needed to effect the desired graphs, provided software such as Mathematica Ⓣ or EXCEL Ⓣ is available.

4.1 Two Modalities

As already mentioned, LABATCH.2 provides two ways of accessing data for statistical analysis. One requires the user to insert a call statement into the data-generating program which executes the call each time it produces a new data vector. Calling and executing BATCH_MEANS each of T times (Table 1) that a data vector with S_NUM entries is generated results in O(S_NUM×T) computing time and O(S_NUM× $\log_2$T) space being used to generate the LABATCH.2 output. Both complexities arise from choosing rules that cause either $B_{j+1} = 2B_j$ or $B_{j+1} \doteq \sqrt{2}B_j$ on successive reviews $j = 1, 2, \ldots$. The space bound is particularly appealing when $t =$T is very large. Yarberry (1993) and Alexopoulos et al. (1997) describe the basis for these complexities.

The other option allows LABATCH.2 to read its data from a file. This feature gives LABATCH.2 a considerably broader range of application than merely for in-line generated sample records. We illustrate how this option works in the context of the M/M/1 example, but stress the applicability of the approach to stored sample data.

If IN_UNIT=30 and OUT_UNIT=15 (Table 1), then a main program needs to call BATCH_MEANS just once to cause LABATCH.2 to read its sample data from a file called c.30 / fort.30 / SIMU30 and to write its output to c.15 / fort.15 / SIMU15. While the O(S_NUM×T) time and O(S_NUM× $\log_2$T) space bounds remain for LABATCH.2, they do not tell the whole story. In particular, the input file requires space proportional to S_NUM×T, which for S_NUM as small as 1 can be substantial for sufficiently large T.

Programs written in C, FORTRAN, or SIMSCRIPT II.5 can implement the second option without qualification. Any program that provides access to and execution of a C, FORTRAN, or SIMSCRIPT II.5 subroutine can take advantage of the first option. In a simulation environment, a user-written program in a language that provides standard linkages for incorporating a subroutine generally will consume less time calling BATCH_MEANS and analyzing data than a program generated at the icon-level in a point-and-click environment.

We now compare timing for the two options for the M/M/1 simulation example using the FORTRAN implementation, which has the standard linkages for subroutine calls.

Table 4: Computing times for M/M/1 example

($t = 10^7$, S_NUM=2, FORTRAN version on a SUN SPARCstation 5)

Case	M/M/1 simulation execution	Repeated in-line call and execution of LABATCH.2	Repeated in-line call but no execution of LABATCH.2	Single call to LABATCH.2	Write sample record to disk	Read sample record from disk	Time (secs.)
1	$\surd$	—	—	—	—	—	170
2	$\surd$	$\surd$	—	—	—	—	591
3	$\surd$	—	$\surd$	—	—	—	200
4	$\surd$	—	—	—	$\surd$	—	1838
5	—	—	—	$\surd$	—	$\surd$	1957

Table 4 shows the computing for selected aspects of execution on a SUN SPARCstation 5. Let T_i denote the computing time for Case i for $i = 1, \ldots, 5$. When switching from a simulation with an in-line call to BATCH_MEANS (Case 2) to a simulation that writes data to disk (Case 4) and makes a single call to BATCH_MEANS to read these data from disk and analyze them (Case 5), Table 4 reveals a $(T_4 + T_5 - T_1)/(T_2 - T_1) = 8.61-$fold increase in computing time. This inflation arises principally from the substantially greater time required to write a data vector to disk and then to read these data, as compared with the computing time required to call BATCH_MEANS. Additional computing experience with LABATCH.2 indicates that this disparity in times for Case 1 versus Cases 4 + 5 grows as S_NUM, T, or both increase. Hence, for S_NUM=1, one would observe a smaller relative disparity in the ratio of times than the one just presented.

It is also of interest to note that for the in-line option

$$\frac{\text{time spent analyzing data}}{\text{time spent calling BATCH\_MEANS}} = \frac{T_2 - T_3}{T_3 - T_1} = 13.0.$$

Since the call merely passes the address of the data vector to BATCH_MEANS, the expended call time is independent of S_NUM. Therefore, this last time ratio would grow as S_NUM increases.

The timings in Table 4 encourage the in-line use of LABATCH.2, whenever possible. However, the capability for applying LABATCH.2 to an extant data file testifies to its versatility in settings other than in-line use.

4.2 Interacting with the Analysis

In queueing simulations, σ_∞^2 usually increases superlinearly with traffic intensity. In Markov chain Monte Carlo sampling, the time to update the state vector grows proportionally to its number of coordinates and σ_∞^2 usually increases superlinearly with this number. This variance inflation can be substantial when applying Gibbs and Hastings Metropolis sampling to lattice topologies. In these settings, LABATCH.2 offers a way of viewing interim results that enables the user to assess the quality of the σ_∞^2 estimates during a run. If the interim results suggest that an acceptable level of accuracy has been achieved, then LABATCH.2 allows the user to terminate the sampling experiment at that point, thus saving computing time.

If SCREEN=0 in expression (2), then LABATCH.2 performs as described in Sec. 1. However, if SCREEN=1, then during execution of a simulation, LABATCH.2 displays on the screen, $\sqrt{BW(L,B)}$, for series i in column $i+1$ for $i = 1, \ldots, \min(7, \text{S\_NUM})$, where screen size dictates the upper limit on i. LABATCH.2 then asks the user if he/she wishes to continue.

Figure 8 shows the displayed tableau for the steady-state M/M/1 simulation, where column 2 shows the sequence of estimates for σ_∞ for Series 1 and column 3 does likewise for σ_∞ for Series 2. If the user concludes that $\sqrt{BW(L,B)}$ has stabilized in each column, so that the systematic errors have become negligible, and if the $\sqrt{BW(L,B)}/\sqrt{\text{No. of obs.}}$ give acceptably small standard errors for the purposes of confidence interval computation, then he/she may terminate the simulation by typing n. This action causes LABATCH.2 to compute confidence intervals for the S_NUM and to write both the final and interim review tableaus (Fig. 1) to file OUT_UNIT. If this action occurs immediately after the on-screen display, for review j, then the final tableau, as well as the interim review tableaus, use L_jB_j observations. If the screen display suggests that systematic error remains, then, provided that $L_jB_j < t$ typing y causes LABATCH.2 to collect additional data, to perform the next review, to display the updated $\sqrt{BW(L,B)}$ on screen, and to ask the user whether or not he/she wishes to continue. If $L_jB_j = t'(t)$ and the user types y, then the simulation goes to completion and LABATCH.2 uses all t=T observations for the sample averages and the first $t'(t)$ observations for the final $BW(L,B)$'s.

In the present setting, it is not unreasonable to conclude that systematic error in $\sqrt{BW(L,B)}$ for Series 1 and 2 have become negligible by review 14. That review reveals estimated standard errors of $203.1/\sqrt{286720} = .3792$ and $1.27/\sqrt{286720} = .002372$ for Series 1 and 2 respectively. If a user feels that these standard errors are acceptable, he/she may terminate the T=10^7 experiment at this point by merely typing n in response to the prompt.

By way of qualification we encourage the reader to recognize that this approach differs substantially from the sequential mean estimation approach of Chow and Robbins (1965), as described in Fishman (1978) and Law and Kelton (1991). Whereas the objective there is to estimate μ to within a specified accuracy with specified probability, the objective of the interactive approach here is to allow a user to assess the quality of variance estimates while the sampling experiment is executing.

Figure 8: LABATCH.2 screen display when SCREEN=1

```
LABATCH.2 INTERIM REVIEW STATISTICAL ANALYSIS


Estimation of Sqrt[sigma^2_infinity] by Sqrt[B*W(L,B)] for Series 1 Through  2
******************************************************************************

         No. of
Review    Obs.       1          2

  1        35    0.4475D+01 0.0000D+00
continue[y/n]? y
  2        70    0.6047D+01 0.0000D+00
continue[y/n]? y
  3       140    0.8680D+01 0.0000D+00
continue[y/n]? y
  4       280    0.1313D+02 0.0000D+00
continue[y/n]? y
  5       560    0.4432D+02 0.9014D+00
continue[y/n]? y
  6      1120    0.5278D+02 0.1007D+01
continue[y/n]? y
  7      2240    0.6196D+02 0.1734D+01
continue[y/n]? y
  8      4480    0.8328D+02 0.1762D+01
continue[y/n]? y
  9      8960    0.8721D+02 0.1289D+01
continue[y/n]? y
 10     17920    0.1130D+03 0.1316D+01
continue[y/n]? y
 11     35840    0.2260D+03 0.1398D+01
continue[y/n]? y
 12     71680    0.2941D+03 0.1581D+01
continue[y/n]? y
 13    143360    0.2304D+03 0.1250D+01
continue[y/n]? y
 14    286720    0.1767D+03 0.1220D+01
continue[y/n]? n
[fish]</home/faculty/gfish/labatch.2>%
```

4.3 Implementations

As mentioned in the introduction, C, FORTRAN, and SIMSCRIPT II.5 versions of LABATCH.2 are available via anonymous ftp. The statement:

```
BATCH_MEANS(IN_UNIT,OUT_UNIT,S_NUM,T,DATA,DELTA,RULE,BETA,L_UPPER,SCREEN)
```

in a C program calls BATCH_MEANS. It requires that a previously defined double S_NUM ×1 array, DATA, contains the latest data entries. Table 2 defines all remaining arguments. If numerical values replace IN_UNIT, OUT_UNIT, S_NUM, T, DELTA, RULE, BETA, L_UPPER, and SCREEN, then no need exists to define these variables in the calling program. This also applies to the FORTRAN and SIMSCRIPT II.5 calls.

The statement:

```
      call BATCH_MEANS(IN_UNIT,OUT_UNIT,S_NUM,T,DATA,
     @          DELTA,RULE,BETA,L_UPPER,SCREEN)
```

in a FORTRAN program calls BATCH_MEANS. It requires that a previously defined double precision S_NUM× 1 array, DATA, contain the latest data entries. The character @ appears in column 6 to indicate a continuation.

The statement:

```
call BATCH_MEANS(IN_UNIT,OUT_UNIT,T,S_NUM,DATA(*),
      DELTA,RULE,BETA,L_UPPER,SCREEN)
```

in a SIMSCRIPT II.5 program calls BATCH_MEANS. In addition to reserving space for the double S_NUM×1 array, DATA, in the main program, a user must also include the statements:

```
define DATA                  as 1-dim double array
define PHI                   as double function
define PHI_QUANTILE          as double function
define STUDENT_T_QUANTILE    as double function
```

in the preamble. See CACI (1983, p.91).

4.4 User-Specified l_1 and b_1

LABATCH.2 allows a user to choose any underscored (l_1, b_1) in Table 2 as the initial number of batches and the inital batch size. To effect this option, the user specifies a sample path length T=$2^\alpha l_1 b_1$ where l_1 and b_1 are the desired values and α is a positive integer. In this case, LABATCH.2 uses 100 percent of the data to computed the final BW(L,B).

4.5 Coding

As mentioned earlier LABATCH.2 is a revision of the LABATCH statistical analysis package. The coding of LABATCH provides users with a considerable number of options. For example, it allows a user to specify whether or not headings are to be printed and for each series it allows a user to specify different significance levels for testing H and different confidence levels for the means.

To reduce the burden of decision making that a potential user faces, these options have been severely curtailed in LABATCH.2. However, a perusal of the code makes clear that the code and data structures that support these wider choices remain in place, but suppressed. This was done to reduce the possibility of error when modifying the code and to leave structures in place that may become useful once again in subsequent revisions of LABATCH.2.

References

[1] Alexopoulos, C., G.S. Fishman, and A.T. Seila (1997). Computational experience with the batch means method, *Proceedings of the 1997 Winter Simulation Conference*, Atlanta.

[2] Blomqvist, N. (1967). The covariance function of the M/G/1 queueing system, Skandinavisk Aktuarietidskrift, **50**, 157-174.

[3] CACI, Inc. (1983). SIMSCRIPT II.5 Reference Handbook, CACI Products Company, La Jolla, California.

[4] Carlstein, E. (1985). The use of subseries for estimating the variance of a general statistic from a stationary sequences, *Ann. Math. Statist.*, **14**, 1171-1179.

[5] Chien, C.H. (1989). Small-sample theory for steady state confidence intervals, Technical Report 37, Department of Operations Research, Stanford University.

[6] Chow, Y.S. and H. Robbins (1965). On the asymptotic theory of fixed-width sequential confidence intervals for the mean, *Ann. Math. Statist.*, **36**, 457-462.

[7] Damerdji, H. (1994). Strong consistency of the variance estimator in steady-state simulation output analysis, *Math. Oper. Res.*, **19**, 494-512.

[8] Fishman, G.S. (1978). *Principles of Discrete Event Simulation*, Wiley, New York.

[9] Fishman, G.S. (1996). *Monte Carlo: Concepts, Algorithms, and Applications*, Springer Verlag, New York.

[10] Fishman, G.S. and L.S. Yarberry (1997). An implementation of the batch means method, *INFORMS J. on Comput.*, **9**, 296-310.

[11] Law, A.M. and W.D. Kelton (1991). *Simulation Modeling and Analysis*, McGraw-Hill, New York.

[12] Goldsman, D. and M.S. Meketon (1986). A comparison of several variance estimators for stationary increment stochastic process, Technical Report, School of Industrial and Systems Engineering, Georgia Institute of Technology.

[13] Philipp, W. and W. Stout (1975). Almost sure invariance principles for partial sums of weakly dependent random variables. *Mem. Am. Math. Soc.*, **2**.

[14] von Neumann, J. (1941). The mean square difference, *Ann. Math. Statist.*, **12**, 153-162.

[15] Yarberry, L.S. (1993). Incorporating a dynamic batch size selection mechanism in a fixed - sample-size batch means procedure, unpublished Ph.D. dissertation, Dept. of Operations Research, University of North Carolina, Chapel Hill.

[16] Young, L.C. (1941). On randomness of order sequences, *Ann. Math. Statist.*, **12**, 293-300.