replications before listing several points and references that I hope are new to some readers.

I enjoyed both papers, but my general preference remains unchanged: Use a single long replication, except in special cases such as parallel computing or when stratified or antithetic initial states happen to be easy to determine. This preference for a single replication is due to its robustness to analyst lack of sophistication or time. Fifteen years ago substantial background, insight and effort were required for simulation and for statistics practitioners to analyze complex problems. Commercial software has blossomed in both fields, allowing relatively naive practitioners to expect something good to happen when they give their problem to the computer. Similarly, one day we will expect software to evaluate posterior distributions with little practitioner insight. The single long replication makes negligible the initial bias, thereby alleviating the difficult initial-data-deletion problem.

Glynn and Heidelberger (1992) and Kelton (1989) are recent additions to the extensive literature that discusses initial deletion of warm-up data and choice of initial states.

Glynn (1987), Whitt (1990) and Damerdji (1991) discuss the choice of number of replications, the extreme cases being a single long run and many short runs.

Smith (1984) discusses Monte Carlo sampling from doubly stochastic Markov chains. The motivation is the need to identify nonredundant constraints in mathematical programming. The methods can be used to sample from a density by sampling uniformly within the region defined by the density and the zero plane. The Hit-and-Run sampler (Belisle, Romeijn and Smith 1992) is a generalization to nonuniform distributions.

Since essentially all point estimators are asymptotically normal, sampling error is well summarized by point-estimator standard error. The method of nonoverlapping adjacent batch means (NBM) is extended to overlapping batch means (OBM) in Meketon and Schmeiser (1984). OBMs are highly dependent, which is acceptable since batches are sufficiently large not when the batch means are essentially independent but (loosely) when each batch subsumes the autocorrelation structure.

Except for end effects, OBM is the Bartlett-window spectral estimator with lag-window length equal to the batch size. Therefore, OBM has the same bias but only two-thirds the variance of NBM. Both NBM and OBM estimator are easily computed in $O(n)$ time; therefore OBM dominates NBM for Markov chain sampling.

For NBM, OBM and some other estimators based on batching, the mse-optimal batch size is asymptotically

$$m^* = \left[2n \left(\frac{c_b^2}{c_v} \right) \left(\frac{\gamma_1}{\gamma_0} \right)^2 \right]^{1/3},$$

where c_b and c_v are the estimator's bias and variance constants, respectively, and γ_1 / γ_0 is the center of gravity of the absolute values of the autocorrelation lags. For OBM $c_b = 1$ and $c_v = 4/3$. Geyer's Theorem 3.1 helps to estimate the autocorrelation center of gravity, which is problem dependent, but the goal is to estimate optimal batch size without estimating individual autocorrelations.

An advantage of batch-means methods is that they extend directly to point estimators that are not means. Schmeiser, Avramidis and Hashem (1990) discuss sufficient assumptions and provide a code for overlapping batch variances and overlapping batch quantiles.

Nelson (1989) quantifies the additional number of batches needed when estimating optimal weights for control variates.

For random-number and random-variate generation, see Fishman and Moore (1986) and Devroye (1986), respectively.

Glasserman (1991) discusses single-replication methods for estimating derivatives of performance measures with respect to system design parameters. Could similar methods be used to estimate the change, for example, in the posterior mean caused by a unit change in the prior mean?

A variety of other simulation-experiment issues are discussed in Schmeiser (1990).

Comment

Luke Tierney

Both papers make some interesting contributions to the discussion of issues related to Markov chain Monte Carlo. Geyer's variance estimates that take advantage of the Markov chain structure appear to be particularly promising and worthy of further investigation. As these methods require a reversible chain, they are not directly applicable to the fixed scan Gibbs sampler. But several simple devices are available for making Gibbs samplers reversible, including random scans,

Luke Tierney is Professor, School of Statistics, University of Minnesota, Vincent Hall, Minneapolis, Minnesota 55455.

random permutation scans, and paired forward and reverse scans.

The paper of Gelman and Rubin makes two main contributions. The first is to propose and motivate a useful numerical criterion for assessing convergence, the potential scale reduction factor. The second is an argument in favor of multiple runs of Markov chain samplers. I would like to discuss each of these in turn.

1. THE CONVERGENCE CRITERION

For the purposes of this discussion, it may be useful to give a simplified outline of the motivation for the convergence criterion used by Gelman and Rubin. The posterior distribution on x is approximately $N(\mu, \sigma^2)$. The posterior mean μ is not known exactly, but the uncertainty about it can be approximated by a $N(\hat{\mu}, \sigma_\mu^2)$ distribution. Using this uncertainty as a prior distribution, we get a predictive distribution for x that has variance $\sigma^2 + \sigma_\mu^2$. If we could eliminate our uncertainty about μ, then this would reduce the width of our intervals for x by the potential scale reduction factor

$$\sqrt{R} \approx \sqrt{1 + \frac{\sigma_\mu^2}{\sigma^2}}.$$

In Gelman and Rubin's development, the uncertainty about μ is due to the fact that μ is being determined by a Markov chain Monte Carlo experiment. But the argument can also be used with other Monte Carlo methods and with non-Monte Carlo methods. For example, my experience with asymptotic approximations suggests that in many reasonable problems, perhaps after some reparameterization, the error in a first-order approximation to the posterior mean should be on the order of 0.2σ or less, and the error in a second-order approximation should be around 0.1σ or less, with σ the posterior standard deviation. If I believe that a particular problem is sufficiently well behaved, then this allows me to compute the approximate scale reductions I could get by using more accurate computational methods as 1.04 and 1.01, respectively.

Turning to Monte Carlo methods, even though it is rarely possible to use i.i.d. sampling from the posterior distribution, it may still be useful to examine the scale reduction factor to determine the number of i.i.d. observations that would be needed to give a satisfactory approximation. In the i.i.d. case with a sample of size N, we have $\sigma_\mu^2 = \sigma^2/N$, and the simplified scale reduction factor is

$$\sqrt{R} \approx \sqrt{1 + \frac{1}{N}} \approx 1 + \frac{1}{2N}.$$

Thus to achieve the accuracy of a second-order approximation in a well-behaved problem would require at least $N = 100$ i.i.d. observations from the posterior distribution.

For non-i.i.d. sampling methods using a total of N observations, σ_μ^2 is usually larger than σ^2/N, though it could be smaller if good variance reduction methods are available. Geweke (1989) defines the *relative numerical efficiency* of a Monte Carlo method as

$$RNE = \frac{\sigma^2/N}{\sigma_\mu^2},$$

the fraction of an i.i.d. observation from the posterior that is equivalent to one observation from the Monte Carlo method. In terms of the RNE, the scale reduction factor is

$$\sqrt{R} \approx \sqrt{1 + \frac{1}{N \times RNE}}.$$

A guess for the RNE can thus be used to adjust the estimate of the sample size needed in a particular problem. Experience to date seems to suggest that in well-behaved problems, the RNE for Gibbs samplers and importance sampling methods may be as large as 20%, but in more difficult problems, it can be well below 1%. Depending on the nature of the problem, this suggests sample sizes of $N = 500$ to $N = 10{,}000$ or more.

Once an experiment has been carried out using a reasonable sample size, more accurate estimates of the scale reduction factor can be constructed and used to determine if additional sampling is needed. This requires estimation of the two variances. With i.i.d. sampling or importance sampling this is fairly straightforward. The dependence structure in Markov chain methods makes estimating σ_μ^2 more difficult but by no means impossible, as Geyer points out. The approach of Gelman and Rubin, again simplified for the purposes of this discussion, involves estimating σ^2 by W and σ_μ^2 by B/n. The use of W seems reasonable, since it is likely to produce an underestimate if n is too small but should be accurate once n is large enough. The use of B seems less satisfactory since it is based on a very small number of degrees of freedom -9 in the recommendation of the paper. Gelman and Rubin do allow for this by using an upper percentile of the distribution of $\sqrt{R}$, but if n is anywhere close to large enough for the individual chains to have reached equilibrium, then it should be possible to do much better by using information from within the chains to estimate σ_μ^2.

As in most simulations involving dependent data, in Markov chain Monte Carlo experiments an initial burn-in period is needed to bring the series close to equilibrium, followed by a considerably longer period to collect enough data on the equilibrium distribution to produce estimates with acceptable variances. The scale reduction factor seems like a useful measure for assessing the adequacy of the length of the equilibrium period. It is related to the relative numerical efficiency

criterion, but it has the nice feature of emphasizing that excessive precision in approximating μ is not of much use if there is reasonable spread in the posterior distribution itself. The argument for using the scale reduction factor, with the particular multiple chain structure and variance estimates proposed in the paper, for determining whether runs are long enough for n to represent an adequate burn-in period seems less persuasive. Given multiple runs, comparing within-sequence means for the second sequence halves to pooled within-sequence variance estimates of the kind outlined by Geyer would seem like a more direct and powerful approach. Even with single-run samplers, the literature on detecting initialization bias (Schruben, 1981, 1982) and on variance estimation in simulation experiments (Goldsman and Schruben, 1990; Goldsman, Meketon and Schruben, 1990) provides useful direct approaches to this problem.

2. MULTIPLE RUNS OR ONE LONG RUN

Multiple runs of a Markov chain Monte Carlo experiment are clearly useful in early exploratory stages, since results from one run can be used for tuning aspects of the next run. Once initial experimentation is complete and a larger experiment is to be used to obtain refined estimates of posterior distributions, the argument for using multiple chains becomes less clear. There appear to be three main reasons put forward for using multiple chains at this stage:

1. To allow easy variance estimation based on independent replications;
2. to reduce correlations in the total sample and to improve the coverage of the parameter space;
3. to aid in detecting problems with the simulation.

The problem with the first argument is that the resulting variance estimates are very inefficient unless the number of chains is quite large, which in turn forces them to be too short. In most problems I would want at least the equivalent of 100 i.i.d. observations from the posterior distribution. Using reasonable single-sequence variance estimation methods it should be possible to extract a variance estimate with the equivalent of about 100 degrees of freedom from that amount of data.

The second point has some merit. Compared to a single run of length $N = mn$ from a Gibbs sampler, say, a series of m independently started runs will be less correlated and produce an overall sample mean with lower variance. But this comes at a price. The length of each run is only n, which may introduce bias if n is too short. To deal with this problem, Gelman and Rubin propose making each run have length $2n$ and discarding the first n observations. Thus the total effort becomes equivalent to a single Gibbs sampler run of length $2mn$, or $(2m - 1)n$ if n observations are also discarded from the Gibbs sampler run. It is no longer clear that the reduction in correlation makes up for the reduction in sample size, though it is possible to construct examples where it will.

But a single run of a Gibbs sampler is not the only single-run alternative to the multiple-run strategy. A hybrid algorithm can be constructed that produces a total of mn observations, uses a Gibbs sampler for most steps, but occasionally, either on a random or a periodic basis, uses a Metropolis independence step (Tierney, 1991) with Gelman and Rubin's t mixture as the candidate generation density. The acceptance tests for these Metropolis steps require the same function evaluations as the SIR algorithm used to select 10 starting points. If the initial density is at all reasonable, then the Metropolis steps should accept at least 5–10% of the time, thus producing a greater reduction in correlation than using 10 separate runs. If the Metropolis steps accept less often, then this indicates that the initial density is not doing a good job and gives a clear warning that the simulation may be in trouble. In addition, since any Markov kernel with the posterior distribution as its invariant distribution will never move a chain farther away from the invariant distribution in total variation distance, incorporating the initial distribution into a single run by such a hybrid method does not destroy, and will often substantially improve, the approach of a long Gibbs sampler run to equilibrium. In summary, by combining the proposed starting strategy with a Gibbs or other Markov chain sampling strategy in a hybrid algorithm, it is possible to build a single-run strategy that dominates both a simple single-run Gibbs sampler and the proposed multiple-run sampler by essentially picking up all the advantages of restarting without any of the disadvantages.

About the only aspect of multiple runs that cannot be incorporated into a single run is the ability to take advantage of a parallel computing environment. While there are pitfalls in running chains on parallel processors, in particular the difficulty of obtaining reliably independent random number streams, parallel environments do offer resources worth exploiting. One approach, currently under investigation, is to try to take advantage of embedded renewal processes that can be found in many Markov chains to produce independent cycles of random length that can be run sequentially or in parallel.