magnitudes of dispersion hyperparameters which are often unknown. As an example, consider the simple balanced, additive, two-way ANOVA model,

$$Y_{ijk} = \mu + \alpha_i + \beta_j + \varepsilon_{ijk}, \quad i = 1, \ldots, I, \quad j = 1, \ldots, J, \; k = 1, \ldots, K,$$

where $\varepsilon_{ijk} \sim N(0, \sigma_e^2)$, $\alpha_i \sim N(0, \sigma_\alpha^2)$, $\beta_j \sim N(0, \sigma_\beta^2)$ and we place a flat prior on μ. Let $\eta_i = \mu + \alpha_i$ and $\rho_j = \mu + \beta_j$, so that η_i centers α_i, and ρ_j centers β_j. Then we can consider four possible parameterizations: (1) μ-α-β; (2) μ-η-β; (3) μ-α-ρ; (4) μ-η-ρ. Gelfand, Sahu and Carlin (1994b) discuss, under varying relative magnitudes for σ_e, σ_α and σ_β, which of these parametrizations is best in terms of mixing (using the diagnostic of Gelman and Rubin, 1992b), which affects the rate of convergence, and in terms of within-chain autocorrelation, which affects the variability of resultant ergodic averages used for inference.

Each of the four parametrizations produces a distinct Gibbs sampler. Following our earlier remarks, we create a fifth MCMC algorithm, which consists of cycling through these four parametrizations in sequence, running one complete single-site updating for each. To keep matters simple, we fix the values of the variance components, set $I = J = K = 5$ and use a sample of data generated from our assumed likelihood. Two interesting cases are shown in Figures 1 and 2, which display monitoring plots, estimated Gelman and Rubin scale reduction factors (labeled "G & R") and lag 1 sample autocorrelations (labeled "acf1") for five initially overdispersed parallel chains of 500 iterations each under the five algorithms. (To conserve space, we show results only for α_1, α_2, β_1, β_2 and μ.) The first figure sets $\sigma_e = 1$, $\sigma_\alpha = 10$ and $\sigma_\beta = 1$, while the second sets $\sigma_e = 1$, $\sigma_\alpha = 10$ and $\sigma_\beta = 20$. In Figure 1, the algorithm based on parametrization #2 (α's centered) is unequivocally the best of the first four, as predicted by the theoretical work in Gelfand, Sahu and Carlin (1994a, b). Matters are less clear in Figure 2, with each of the individual parametrizations having problems with one or more of the parameters. Notice that in both figures, for each component of the parameter space, the fifth algorithm achieves mixing which is as good as that of *any* of the first four. In fact, in Figure 2, the behavior of μ is satisfactory *only* for this composite algorithm. Note also, however, that the lag 1 autocorrelations for the fifth algorithm are fairly high, arising as weighted averages of those from the first four, so the corresponding samples must be used carefully in computing expectations via Monte Carlo integration.

Hence with regard to convergence, in using deterministic cycling through a medley of transition kernels, the analyst is able to achieve the benefits of each (and possibly more) without having to identify their relative quality. The computational effort in switching transition kernels in our examples only requires changing from one linear parametrization to another, and thus is quite efficient. Lastly, in situations where Metropolis steps are to be used within Gibbs samplers, thus necessitating proposal densities, adaptive adjustment of the dispersion of these proposals can be implemented concurrently with the deterministic switching of transition kernels.

ACKNOWLEDGMENTS

The work of the first-named author was supported in part by NSF Grant DMS-93-01316, while the work of the second-named author was supported in part by National Institute of Allergy and Infectious Diseases (NIAID) FIRST Award 1-R29-AI33466.

Comment

Charles J. Geyer

The authors are to be congratulated on this very nice paper, a tour de force in which all of various aspects of MCMC are completely mastered. I find myself largely in agreement with everything in this paper. What comments I have are not really disagreements but mere differences in emphasis.

Charles J. Geyer is Assistant Professor, School of Statistics, University of Minnesota, Minneapolis, Minnesota 55455.

SEPARATION OF CONCERNS

Let me begin my comments with a digression. Dijkstra (1976) in his seminal book on formal analysis of the correctness of computer programs introduces the notion of "separation of concerns." In computing we have "the mathematical concerns about correctness [of algorithms and programs implementing them] and the engineering concerns about execution [speed, memory requirements, user-friendliness, featurality]" and these should be kept separate. There is no point in worrying about speed