Accelerating Computation in Markov Random Field Models for Spatial Data via Structured MCMC

By MURALI HARAN, JAMES S. HODGES, AND BRADLEY P. CARLIN

Division of Biostatistics, School of Public Health, University of Minnesota, Mayo Mail Code 303, Minneapolis, Minnesota 55455–0392, U.S.A.

Correspondence author: Bradley P. Carlin
telephone: (612) 624-6646
fax: (612) 626-0660
email: brad@muskie.biostat.umn.edu

May 9, 2002

Acknowledgments

The first and third authors were supported in part by NIAID grant R01-AI41966 and NSF/EPA grant SES 99-78238, while the second author was supported in part by the Minnesota Oral Health Clinical Research Center, NIH/NIDR grant P30-DE09737.

Accelerating Computation in Markov Random Field Models for Spatial Data via Structured MCMC

Summary

Spatial Poisson models for areal count data of the kind proposed by Besag, York and Mollie (1991) use non-stationary "intrinsic autoregressions," also often referred to as "conditionally autoregressive" (CAR) models. Bayesian inference for these models has generally involved using single parameter updating Markov chain Monte Carlo algorithms, which often exhibit slow mixing (i.e. poor convergence) properties. These spatial models are richly parameterized and lend themselves to the Structured Markov Chain Monte Carlo (SMCMC) algorithms introduced by Sargent, Hodges and Carlin (2000). SMCMC provides a simple, general and flexible framework for accelerating convergence in an MCMC sampler by providing a systematic way to block groups of similar parameters while taking full advantage of the posterior correlation structure induced by the model and data. Among the SMCMC strategies we consider here are blocking using different size blocks (grouping by geographical region), reparameterization, updating jointly with and without model hyperparameters, "oversampling" some of the model parameters, and "pilot adaptation" versus continuous tuning techniques for the proposal density. We apply the techniques presented here to datasets on cancer mortality and late detection in the state of Minnesota. We find that, compared to univariate sampling procedures, our techniques will typically lead to more accurate posterior estimates, and they are sometimes also far more efficient in terms of the number of effective samples generated per second.

KEY WORDS: Blocking; Convergence acceleration; Disease mapping; Gibbs sampling; Hierarchical model; Hierarchical centering; Metropolis-Hastings algorithm; Reparameterization.

1 Introduction

Structured Markov Chain Monte Carlo (SMCMC) was introduced by Sargent, Hodges and Carlin (2000) as a general method for Bayesian computing in richly-parameterized models. Here, "richly parameterized" refers to hierarchical and other multilevel models. SMCMC provides a simple, general and flexible framework for accelerating convergence in an MCMC sampler by providing a systematic way to update groups of similar parameters in blocks while taking full advantage of the posterior correlation structure induced by the model and data. Sargent et al. (2000) apply SMCMC to several different models, including a hierarchical linear model with normal errors and a hierarchical Cox proportional hazards model.

Blocking, i.e., simultaneously updating multivariate blocks of (typically highly correlated) parameters, is a general approach to accelerating MCMC convergence. Liu (1994) and Liu et al. (1994) confirm its good performance for a broad class of models, though Liu et al. (1994, Sec.5) and Roberts and Sahu (1997, Sec 2.4) give examples where blocking slows a sampler's convergence. In this paper, we show that spatial models of the kind proposed by Besag, York and Mollie (1991) using non-stationary "intrinsic autoregressions" are richly parameterized and lend themselves to the SMCMC algorithm. Bayesian inference via MCMC for these models has generally used single parameter updating algorithms with often poor convergence and mixing properties. There have been some recent attempts to use blocking schemes for similar models. Cowles (2002a,b) uses SMCMC blocking strategies for geostatistical and areal data models with normal likelihoods, while Knorr-Held and Rue (2002) implement blocking schemes using algorithms that exploit the sparse matrices that arise out of the areal data model.

We study several different strategies for block-sampling parameters in the posterior distribution when the likelihood is Poisson. Among the SMCMC strategies we consider here are blocking using different-sized blocks (grouping by geographical region), updating jointly with and without model hyperparameters, "oversampling" some of the model parameters, reparameterization via hierarchical centering and "pilot adaptation" of the transition kernel. Our results suggest that our techniques will generally be far more accurate (produce less correlated samples) and often more efficient (produce more effective samples per second) than univariate sampling procedures.

The remainder of our paper is organized as follows. Section 2.1 provides the details of our spatial models for areal data. The rest of Section 2 then lays out the basics of the constraint case formulation, and shows how our class of spatial models can be formatted so the SMCMC approach can be used to advantage. A variety of different SMCMC algorithms are outlined in Section 3, where we describe the features and potential advantages and disadvantages of each. Section 4 then presents our two datasets (both related to cancer control in the counties of the state of Minnesota), followed by detailed tabular and graphical results concerning the efficiency of our SMCMC algorithms relative to each other and to the univariate updating algorithm. Our datasets are quite different in size and character, and are suggestive of general situations where SMCMC may or may not be expected to pay significant dividends. Finally, Section 5 discusses our findings, briefly compares them with other similar approaches for areal data, and offers directions for future applied and methodologic research in this area.

2 Applying Structured MCMC to Areal Data

2.1 Spatial Modeling of Areal Data

Besag, York and Mollie (1991) describe a spatial model for areal data (i.e., data arising as sums or averages over geographic regions). The number of disease events in region i, Y_i, is modeled as a Poisson random variable with mean $E_i \exp(\mu_i)$. E_i is the expected number of events in region i while μ_i is the log-relative risk of disease. So $Y_i \sim Po(E_i e^{\mu_i})$, with μ_i modeled linearly as

$$\mu_i = \theta_i + \phi_i \ , \ i = 1, ..., N,$$

where N is the total number of regions, and $\{\theta_1, .., \theta_N\}, \{\phi_1, .., \phi_N\}$ are vectors of random effects. The θ_i's are independent and identically distributed Gaussian normal variables, while the ϕ_i's are conditionally autoregressive and assumed to follow a Gaussian Markov random field (GMRF). In this way, each θ_i captures the ith region's extra-Poisson variability due to area-wide heterogeneity, while each ϕ_i captures the ith region's excess variability attributable to regional clustering. These distributions are specified as follows:

$$\theta_i|\tau_h \sim N(0, 1/\tau_h), \text{ and } \phi_i|\phi_{j\neq i} \sim N(\mu_{\phi_i}, \sigma^2_{\phi_i}), i = 1, ..., N ,$$

$$\text{where } \mu_{\phi_i} = \frac{\sum_{j\neq i} w_{ij}\phi_j}{\sum_{j\neq i} w_{ij}} \text{ and } \sigma^2_{\phi_i} = \frac{1}{\tau_c \sum_{j\neq i} w_{ij}}.$$

The μ_{ϕ_i} for a region i is thus a weighted average of the clustering parameters in other regions. Most commonly, w_{ij} is taken as 0 unless regions i and j are immediate neighbors. If the regions i and j are adjacent, w_{ij} is set to 1, although since they are merely weights, other forms of the w_{ij} are also possible. Note that the prior on ϕ_i leaves the overall level of the GMRF unspecified; the prior is therefore improper due to translation invariance.

A question of epidemiological interest is whether more of the variability of the observations in different regions is captured by heterogeneity (corresponding to global variability) or by clustering (corresponding to local variability). Thus the variance components of the θ_i and the ϕ_i are of interest in their own right (Best et al. , 1999), suggesting the need for hyperpriors on the τ_h and τ_c. A "fair" specification here is complicated by the fact that τ_h is an *unconditional* prior precision, while τ_c is part of a *conditional* prior precision. We place conjugate Gamma hyperpriors on the precision parameters, namely $\tau_h \sim G(\alpha_h, \beta_h)$ and $\tau_c \sim G(\alpha_c, \beta_c)$ with $\alpha_h = 1.0$, $\beta_h = 100.0$, $\alpha_c = 1.0$ and $\beta_c = 50.0$ (these hyperpriors have means of 100 and 50, and standard deviations of 10000 and 2500 respectively, a specification recommended by Bernardinelli et al., 1995). See Eberly and Carlin (2000) for more discussion of "fair but vague" priors for τ_h and τ_c.

2.2 SMCMC Algorithm Basics

Following Hodges (1998), we consider a hierarchical model expressed in the general form

$$\left[\begin{array}{c} y \\ \hline 0 \\ \hline M \end{array}\right] = \left[\begin{array}{c|c} X_1 & 0 \\ \hline H_1 & H_2 \\ \hline G_1 & G_2 \end{array}\right] \left[\begin{array}{c} \theta_1 \\ \hline \theta_2 \end{array}\right] + \left[\begin{array}{c} \epsilon \\ \hline \delta \\ \hline \xi \end{array}\right]. \qquad (1)$$

The first set of rows of this layout correspond to the "data cases," or the terms in the joint posterior into which the response, the data y, enters directly. The second set of rows (corresponding to the H_i) are called "constraint cases" since they place stochastic constraints on possible values of θ_1 and θ_2. The third set of rows, the "prior cases" for the model parameters, have known (specified) error variances for these parameters. Equation (1) can be expressed as $Y = \mathrm{X}\mathbf{\Theta} + \mathrm{E}$, where X and Y are known, $\mathbf{\Theta}$ is unknown, and E is an error term with block diagonal covariance matrix $\Gamma = \mathrm{Diag}(\mathrm{Cov}(\epsilon), \mathrm{Cov}(\delta), \mathrm{Cov}(\xi))$. If the error structure for the data is normal, i.e., if the ϵ vector in the constraint case formulation (1) is normally distributed, then the conditional posterior density of $\mathbf{\Theta}$ is

$$\Theta | Y, \Gamma \sim N((X^T \Gamma^{-1} X)^{-1}(X^T \Gamma^{-1} Y)\ ,\ (X^T \Gamma^{-1} X)^{-1})\ . \tag{2}$$

The basic SMCMC algorithm is then nothing but the following two-block Gibbs sampler :

(1) Sample $\mathbf{\Theta}$ as a single block from the above normal distribution, using the current value of Γ.

(2) Update Γ using the conditional distribution of the variance components with the current value of $\mathbf{\Theta}$.

In our spatial model setting, the errors are not normally distributed, so the normal density described above is not the correct conditional posterior distribution for $\mathbf{\Theta}$. Still, a SMCMC algorithm with a Metropolis-Hastings implementation can be used, with the normal density in (2) taken as the candidate density.

2.3 Application to Spatial Modeling

Consider a data set of N regions with C pairs of adjacent neighbors. Thus, there are $2N+2$ model parameters: $\{\theta_i : i = 1, \ldots N\}$, $\{\phi_i : i = 1, \ldots N\}$, τ_h and τ_c. The SMCMC algorithm requires that we transform the Y_i data points to $\hat{\mu}_i = \log(Y_i/E_i)$, which can be conveniently thought of as the response since they should be roughly linear in the model parameters (the θ_i's and ϕ_i's). For the constraint case formulation, the different levels of the model are written down case by case. The data cases are $\hat{\mu}_i$, $i = 1, \ldots, N$. The constraint cases for the θ_i's are $\theta_i \sim N(0, 1/\tau_h)$, $i = 1, \ldots, N$. For the constraint cases involving the ϕ_i's, the differences between the neighboring ϕ_i's can be used to get an unconditional distribution for the ϕ_i's using pairwise differences (Besag et al. 1995). Thus the constraint cases can be written as

$$(\phi_i - \phi_j)|\tau_c \sim N(0, 1/\tau_c) \text{ for each } i, j \text{ that are adjacent regions.} \tag{3}$$

To obtain an estimate of Γ, we need estimates of the variance-covariance matrix corresponding to the $\hat{\mu}_i$'s (the data cases) and initial estimates of the variance-covariance matrix for the constraint cases (the rows corresponding to the θ_i's and ϕ_i's). Using the delta method, we can obtain an approximation as follows : assume $Y_i \sim N(E_i e^{\mu_i}, E_i e^{\mu_i})$ (roughly), so invoking the delta method we can see that Var$(log(Y_i/E_i))$ is approximately $1/Y_i$. A reasonably good starting value is particularly important here since we never update these variance estimates (the data variance section of Γ stays the same throughout the algorithm). For initial estimates of the variance components corresponding to the θ_i's and the ϕ_i's, we can use the mean of the hyperprior densities on τ_h and τ_c, and substitute these values into Γ.

The SMCMC candidate generating distribution is thus of the form (2), with the Y_i's replaced by $\hat{\boldsymbol{\mu}}$. To be able to compute the Hastings ratio, the distribution of the ϕ_i's is rewritten in the joint pairwise difference form (Besag et al. , 1995), with the appropriate exponent for τ_c (Hodges and Carlin, 2001):

$$p(\phi_1, \phi_2, ..., \phi_N | \tau_c) \propto \tau_c^{(N-1)/2} \exp\left\{ -\frac{\tau_c}{2} \sum_{i \sim j} (\phi_i - \phi_j)^2 \right\}, \tag{4}$$

where $i \sim j$ if i and j are neighboring regions. Finally, the joint distribution of the θ_i's is given by

$$p(\theta_1, \theta_2, ..., \theta_N | \tau_h) \propto \tau_h^{N/2} \exp\left\{ -\frac{\tau_h}{2} \sum_{i=1}^{N} \theta_i^2 \right\}. \tag{5}$$

As described above, the response vector is $\hat{\boldsymbol{\mu}}^T = \{log(Y_1/E_1), log(Y_2/E_2), ..., log(Y_N/E_N)\}$. The $(2N + C) \times 2N$ design matrix for the spatial model is defined by :

$$X = \left[\begin{array}{c|c} I_{N\times N} & I_{N\times N} \\ \hline -I_{N\times N} & 0_{N\times N} \\ \hline 0_{C\times N} & A_{C\times N} \end{array} \right]. \tag{6}$$

The design matrix is divided into two halves, the left half corresponding to the N θ_i's and the right half referring to the N ϕ_i's. The top section of this design matrix is a $N \times 2N$ matrix relating $\hat{\mu}_i$ to the model parameters θ_i and ϕ_i. In the ith row, a 1 appears in the ith and $(N+i)$th columns while 0s appear elsewhere. Thus the ith row corresponds to $\mu_i = \theta_i + \phi_i$. The middle section of the design matrix is an $N \times 2N$ matrix which imposes a stochastic constraint on each θ_i separately (θ_i's are i.i.d normal). The bottom section of the design matrix is a $C \times 2N$ matrix with each row having a -1 and 1 in the $(N+k)$th and $(N+l)$th columns respectively, corresponding to a stochastic constraint being imposed on $\phi_l - \phi_k$ (using the pairwise difference form of the prior on the ϕ_i's as described in (3) with regions l and k being neighbors). The variance-covariance matrix Γ is a diagonal matrix with the top left section corresponding to the variances of the data cases, i.e., the $\hat{\mu}_i$'s. Using the variance approximations described above, the $(2N+C) \times (2N+C)$ block diagonal variance-covariance matrix is

$$\Gamma = \left[\begin{array}{c|c|c} Diag(1/Y_1, 1/Y_2, \ldots, 1/Y_N) & 0_{N\times N} & 0_{N\times C} \\ \hline 0_{N\times N} & \frac{1}{\tau_h} I_{N\times N} & 0_{N\times C} \\ \hline 0_{C\times N} & 0_{C\times N} & \frac{1}{\tau_c} I_{C\times C} \end{array}\right]. \tag{7}$$

Note that the exponent on τ_c in (4) would actually be $C/2$ (instead of $(N-1)/2$) if obtained by taking the product of the terms in (3). Thus, (3) is merely a form we use to describe the distribution of the ϕ_is for our constraint case specification. The formal way to incorporate the distribution of the ϕ_is in the constraint case formulation is by using an alternate specification of the joint distribution of the ϕ_is, as described in Besag and Kooperberg (1995). This form is a $N \times N$ Gaussian density with precision matrix, Q,

$$p(\phi_1, \phi_2, ..., \phi_N | \tau_c) \propto \exp\left(-\frac{\tau_c}{2} \boldsymbol{\phi}^T Q \boldsymbol{\phi}\right), \text{ where } \boldsymbol{\phi}^T = (\phi_1, \phi_2, ..., \phi_N), \tag{8}$$

and

$$Q_{ij} = \begin{cases} c & \text{if } i = j \text{ where } c = \text{ number of neighbors of region } i \\ 0 & \text{if } i \text{ is not adjacent to } j \\ -1 & \text{if } i \text{ is not adjacent to } j \end{cases}$$

However, it is possible to show that this alternate formulation (using the corresponding design and Γ matrices) results in the same SMCMC candidate mean and covariance matrix for $\mathbf{\Theta}$ given τ_h and τ_c as the one described in (2); see Haran (2002) for details.

3 Algorithmic Schemes

Univariate MCMC (UMCMC): For the purpose of comparison with the different blocking schemes, we began with a univariate (updating one variable at a time) sampler. This was done by sampling τ_h and τ_c from their gamma full conditional distributions, and then, for each i, sampling each θ_i and ϕ_i from its full conditional distribution. The latter used a Metropolis step with univariate Gaussian random walk proposals, the variances of which were tuned to produce acceptance rates between 30% and 70%.

Reparameterized Univariate MCMC (RUMCMC): We reparameterized from $(\theta_1, \ldots, \theta_N, \phi_1, \ldots, \phi_N)$ to $(\mu_1, \ldots, \mu_N, \phi_1, \ldots, \phi_N)$, where $\mu_i = \theta_i + \phi_i$. The (new) model parameters and the precision parameters were sampled in a similar manner as for UMCMC. This "hierarchical centering" was suggested by Besag et al. (1995) and Waller et al. (1997) for the spatial model, and discussed in general by Gelfand et al. (1995).

Structured MCMC (SMCMC): We initially tried a pilot adaptation, which involved sampling (τ_h, τ_c) from their gamma full conditionals, updating the Γ matrix using the averaged (τ_h, τ_c) sampled so far, updating the SMCMC candidate covariance matrix and mean vector using the Γ matrix, and then sampling $(\boldsymbol{\theta}, \boldsymbol{\phi})$ using the SMCMC candidate in a Metropolis-Hastings step. We ran the above steps for a "tuning" period, after which we fixed the SMCMC candidate mean and covariance, sampled (τ_h, τ_c) as before, and used the Metropolis-Hastings to sample $(\boldsymbol{\theta}, \boldsymbol{\phi})$ using SMCMC proposals. Some related strategies we tried included adaptation of the Γ matrix more or less frequently, adaptation over shorter and longer period of time, and pilot adaptation while blocking on groups of regions.

Our experience with pilot adaptation schemes indicated that a single proposal, regardless of adaptation period length, will probably be unable to provide a reasonable acceptance rate for the many different values of (τ_h, τ_c) that will be drawn in realistic problems. As such, we turned to oversampling $\mathbf{\Theta}$ relative to (τ_h, τ_c); that is, the SMCMC proposal is always based on the current (τ_h, τ_c) value. In this algorithm, we sample τ_h and τ_c from their gamma full conditionals, and then compute the SMCMC proposal based on the Γ matrix using the generated τ_h and τ_c. For each (τ_h, τ_c) pair, we run a Hastings independence subchain by sampling a sequence of length 100 (say) of $\boldsymbol{\Theta}$'s using the SMCMC proposal. Further implementational details for this algorithm are given in Haran (2002).

Reparameterized Structured MCMC (RSMCMC): This algorithm is the SMCMC analogue of the reparameterizated univariate algorithm (RUMCMC). The algorithm follows exactly the same steps as the SMCMC algorithm, with the only difference being that $\boldsymbol{\Theta}$ is now $(\boldsymbol{\mu}, \boldsymbol{\phi})$ instead of $(\boldsymbol{\theta}, \boldsymbol{\phi})$, and the proposal distribution is adjusted according to the new parameterization. It is obtained in the same manner as before, as described in Section 2.3.

Space prevents a full description of the many other algorithms we investigated, including ones which block parameters based on geographical proximity, and those that block the precision parameters together with $\boldsymbol{\Theta}$. Again, see Haran (2002) for details.

4 Results

4.1 Description of Datasets

4.1.1 Minnesota Cancer Detection Data

Our first dataset is taken from the Minnesota Cancer Surveillance System (MCSS), a cancer registry maintained by the Minnesota Department of Health. The MCSS is population-based for the state of Minnesota, and collects information on geographic location and stage at detection for colorectal, prostate, lung, and female breast cancers. An external audit (Cancer Surveillance and Control Program, 1997) performed in June 1996 estimated that MCSS hospital-based case finding was 99.6% complete for microscopically confirmed cancers, and 99.1% complete for all cancers. We may thus think of these data as an essentially complete picture of all cancers that occur in Minnesota.

We illustrate our computational approaches by analyzing the MCSS data for two of the cancers, breast and colorectal. Each of the 87 counties in the data set has associated with it the total number of cancer cases recorded between 1995 and 1997, and the number of these detected late. We then take the expected number of late detections for that county as the number of cancer cases for that county multiplied by the statewide rate of late detections. The question of interest is whether there are clusters of counties in the state of Minnesota with much higher than expected late detection rates for either cancer. The spatial model provides smoothed estimates of the relative risk of cancer cases being detected late in each county. Counties (or clusters of counties) emerging with higher smoothed late detection rate may be targets for aggressive screening efforts, such as the deployment of mobile mammography units.

4.1.2 Minnesota Cancer Mortality Data

Our second dataset comes from the Minnesota Department of Health's Center for Health Statistics, and consists of the age-adjusted cancer death rates r_i^* for each county i during the period 1991-1998. Deaths from cancer were determined by the ICD-9 codes on the death certificates of Minnesota residents. Census data from the same period were used to obtain an average population n_i for each county, thus determining an "age-adjusted cancer death total," $Y_i = n_i r_i^*$. Similar to the previous subsection, we then specify an expected number of age-adjusted deaths for each county as $E_i = n_i R$, where $R = (\sum_i Y_i)/(\sum n_i)$, the statewide age-adjusted cancer death rate. The model of Section 3 now applies as before. Again the substantive problem of interest is to determine overall spatial trends in cancer death, and whether or not counties with significantly elevated smoothed rates exist.

While the adjacency structure remains as in the above two data sets, the counts in this data set are appreciably higher, with a mean count around 700 (versus 34 and 51 for breast and colorectal cancer detection, respectively) and a lowest count of 48 (versus 2 and 6). This dataset thus affords an opportunity to study the performance of our algorithms operating on a dataset with larger counts per region.

4.2 Summary of Results

4.2.1 Minnesota Cancer Detection Data

For the sake of brevity, we display results for only the four major algorithms described in Section 3, and for only a few parameters for each data set. To make a fair comparison among the various implementations, we use the notion of effective sample size, or ESS (Kass et al. , 1998). ESS is defined for each parameter as the number of MCMC samples drawn divided by the parameter's so-called autocorrelation time, $\kappa = 1 + 2\sum_{k=1}^{\infty} \rho(k)$, where $\rho(k)$ is the autocorrelation at lag k. We estimate κ from the MCMC chain, using the initial monotone positive sequence estimator as given by Geyer (1992). This estimator is protected against random noise, and under some mild assumptions, is a consistent estimator of autocorrelation time.

In what follows, we use ESS as a measure of algorithm accuracy (since less correlated samples lead to more accurate inference) and effective samples per second (ES/s) as a measure of algorithm efficiency. All algorithms were programmed in `C` and run on the same LINUX machines. The parameters presented in the table were selected on the basis of their ESS's. In particular, we select the parameters that have the lowest, median, and highest differences in ESS between the UMCMC and SMCMC algorithm.

When experimenting with simulated data sets, we found that we could get reasonable acceptance rates (over 30%) using pilot adaptation for problems of dimension less than 30, but these acceptance rates quickly dropped as we increased the dimension of the problem. This suggests that it would make sense to have a single normal proposal for *each* generated value of $(\tau_h,\ \tau_c)$, so that the SMCMC proposal and the τ_h and τ_c values are "synchronized." This led to the oversampled SMCMC scheme described in the previous section. Fixing $(\tau_h,\ \tau_c)$ and then immediately tuning the SMCMC proposal to produce an independence subchain of samples from the $\boldsymbol{\Theta}$ posterior helps the sampler achieve high acceptance rates and also good mixing properties (see Tables 1 and 2). For instance, in Table 1, the autocorrelations for the samples for all the model parameters are much lower for SMCMC than for the univariate sampler (UMCMC), and the corresponding ESS is much higher. The same is also true in Table 2, where the samples from the block samplers have practically no autocorrelations, and hence, have high ESS's. Even in terms of ES/s, SMCMC is pretty close to the corresponding univariate schemes for both data sets, with SMCMC doing better than UMCMC for around 50% of the parameters.

The results for the reparameterized versions of these two algorithms, RUMCMC and RSMCMC, suggest that reparameterization is not particularly effective for these two data sets. In fact, RUMCMC does worse than UMCMC for several of these parameters. The SMCMC version of the reparameterization algorithm (RSMCMC) does appear to do quite well, and produces comparable ESSs to the SMCMC algorithm. This suggests that block sampling in the manner we suggest can improve a sampler even under a parameterization that performs poorly for univariate updating schemes.

In summary, for the (relatively small) breast cancer and colorectal cancer detection data sets, ESS is

	Method	mean	sd	AC1	AC5	AC10	ESS	ES/s
ϕ_7	UMCMC	0.011	0.059	0.85	0.56	0.42	2558.52	2.06
ϕ_7	SMCMC	0.013	0.059	0.05	0.01	0	87324.52	6.3
ϕ_7	RUMCMC	0.014	0.059	0.53	0.33	0.22	7613.83	6.06
ϕ_7	RSMCMC	0.012	0.058	0.05	0	0.01	85412.01	6.45
ϕ_{15}	UMCMC	0.065	0.072	0.86	0.62	0.51	1770.87	1.42
ϕ_{15}	SMCMC	0.065	0.071	0.13	0.07	0.07	46110.47	3.33
ϕ_{15}	RUMCMC	0.066	0.071	0.63	0.44	0.32	4436.28	3.53
ϕ_{15}	RSMCMC	0.065	0.069	0.12	0.06	0.06	46418.03	3.51
θ_{56}	UMCMC	-0.048	0.067	0.29	0.06	0.04	34158.58	27.48
θ_{56}	SMCMC	-0.05	0.068	0.14	0.1	0.09	42524.81	3.07
θ_{56}	RUMCMC	-0.049	0.068	0.45	0.12	0.05	22295.57	17.74
θ_{56}	RSMCMC	-0.051	0.068	0.13	0.09	0.1	53139.88	4.01
τ_h	UMCMC	263.305	135.471	-0.02	0.01	0.01	1000	0.8
τ_h	SMCMC	263.213	143.781	0.39	0.02	0.05	464.63	0.03
τ_h	RUMCMC	257.536	127.679	0.01	-0.04	0.02	1000	0.8
τ_h	RSMCMC	253.287	134.534	0.37	-0.02	0.03	505.27	0.04
τ_c	UMCMC	109.862	61.903	0.15	-0.05	0	773.02	0.62
τ_c	SMCMC	109.883	62.774	0.45	0.04	-0.04	353.41	0.03
τ_c	RUMCMC	106.315	59.725	0.06	0.05	-0.02	885.35	0.7
τ_c	RSMCMC	110.55	60.444	0.42	-0.01	0.01	418.74	0.03

Table 1: Selected results for the Minnesota breast cancer detection data set. Each chain was run for 1 million iterations. Only every 10th sample for the θ_i and ϕ_i (and every 1000th sample for τ_h and τ_c, due to the oversampling in SMCMC and RSMCMC) was saved, so the above lag1, lag5 and lag10 ACs are for this thinned sample.

often higher. For both data sets, the block samplers beat the univariate samplers in terms of ESS for practically all the parameters. Even in terms of ES/s, the block samplers are comparable to the much faster univariate samplers, suggesting that the block sampler approaches may be generally preferable to the univariate algorithms.

4.2.2 Minnesota Cancer Mortality Data

To a significant extent, the performance of our SMCMC schemes hinges on the accuracy of the normal approximation to our Poisson likelihood. Since we know that this approximation is better for higher counts, the Minnesota cancer mortality data set described in Subsection 4.1.2 should result in better efficiencies than those observed in the previous subsection. We therefore ran the UMCMC, SMCMC, RUMCMC and RSMCMC schemes on this data set. As expected, UMCMC performs poorly for these large counts, though the reparameterized univariate algorithm (RUMCMC) does provide a significant improvement in this case.

	Method	mean	sd	AC1	AC5	AC10	ESS	ES/s
ϕ_{75}	UMCMC	-0.011	0.06	0.84	0.54	0.4	3144.41	2.53
ϕ_{75}	SMCMC	-0.009	0.06	0	0	0	99207.1	7.16
ϕ_{75}	RUMCMC	-0.011	0.061	0.54	0.33	0.2	7918.18	6.3
ϕ_{75}	RSMCMC	-0.009	0.06	0	0	0	100000	7.55
θ_{69}	UMCMC	-0.017	0.051	0.3	0.16	0.11	16195.42	13.03
θ_{69}	SMCMC	-0.019	0.051	0.03	0.02	0.02	88135.5	6.36
θ_{69}	RUMCMC	-0.018	0.051	0.33	0.11	0.05	24610.39	19.58
θ_{69}	RSMCMC	-0.018	0.051	0.02	0.02	0.01	88732.19	6.7
θ_3	UMCMC	-0.05	0.062	0.25	0.05	0.02	41067.86	33.04
θ_3	SMCMC	-0.051	0.063	0.12	0.11	0.1	42327.36	3.06
θ_3	RUMCMC	-0.05	0.062	0.47	0.14	0.06	19442.6	15.47
θ_3	RSMCMC	-0.051	0.062	0.11	0.1	0.09	61166.16	4.62
τ_h	UMCMC	312.323	139.511	-0.04	0.03	0.01	1000.0	0.87
τ_h	SMCMC	306.829	146.865	0.32	0.02	0.07	502.76	0.04
τ_h	RUMCMC	308.54	138.365	-0.01	-0.08	-0.02	1000	0.8
τ_h	RSMCMC	304.573	140.563	0.27	-0.03	-0.02	595.29	0.04
τ_c	UMCMC	136.488	65.949	0.1	0.01	0.03	811.67	0.65
τ_c	SMCMC	130.292	60.755	0.35	0.04	-0.04	454.9	0.03
τ_c	RUMCMC	131.086	62.205	0.03	-0.02	0.01	940.56	0.75
τ_c	RSMCMC	131.894	62.717	0.36	0.01	0.01	473.75	0.04

Table 2: Selected results for the Minnesota colorectal cancer detection data set. Each chain was run for 1 million iterations. Only every 10th sample for the θ_i and ϕ_i (and every 1000th sample for τ_h and τ_c, due to the oversampling in SMCMC and RSMCMC) was saved, so the above lag1, lag5 and lag10 ACs are for this thinned sample.

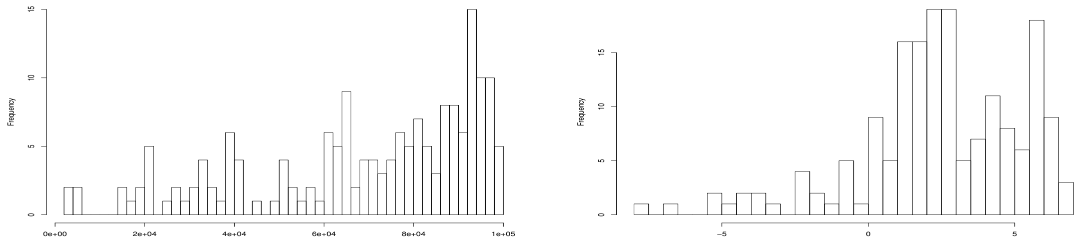
However, SMCMC and RSMCMC still perform better than both univariate algorithms. For a representative set of parameters (selected in the same way as before), Table 3 shows that even when accounting for the amount of time taken by the SMCMC algorithm (in terms of ES/s), the SMCMC scheme results in a far more efficient sampler than the univariate algorithm; for some parameters (not shown here), SMCMC can produce as much as 64 times more effective samples per second.

In fact, the improvement offered by SMCMC is even greater than suggested by Table 3. This can be seen in Figure 1, which shows the differences in ESS for all the parameters, along with the difference in ES/s. The UMCMC algorithm ESSs are always below 16,000 while over 79% of all the ESSs for the SMCMC algorithm are over 50,000, and SMCMC does better than UMCMC for *all* the parameters. Note that, even in terms of ES/s, our SMCMC algorithm improves on the univariate algorithm for all but 13% of the model parameters.

The RSMCMC algorithm generally outperforms RUMCMC in terms of ESS, as can be seen from Table 4 and Figure 2(a). However, Figure 2(b) shows that RUMCMC always has lower ES/s than RSMCMC. Thus,

	UMCMC ESS	SMCMC ESS	diff	ratio	UMCMC ES/s	SMCMC ES/s	diff	ratio
θ_{71}	853.08	100000	99146.92	117.22	0.69	7.22	6.53	10.52
ϕ_{13}	940.32	77742.14	76801.82	82.68	0.76	5.61	4.85	7.42
θ_{27}	59.57	2558.03	2498.46	42.94	0.05	0.18	0.14	3.85

Table 3: Selected results for Minnesota cancer mortality data set. Each chain was run for 1 million iterations.



(a) SMCMC ESS – UMCMC ESS

(b) SMCMC ES/s – UMCMC ES/s

Figure 1 : Comparisons of ESS and ES per second, UMCMC versus SMCMC (positive values favor SMCMC).

the high overhead of blocking seems to limit the cost-effectiveness of RSMCMC in this case.

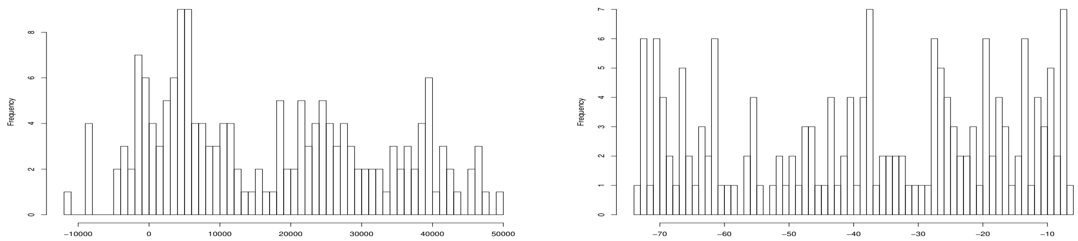
5 Discussion

We have described several block updating algorithms for analyzing Bayesian hierarchical models for disease mapping as described in Section 2.1. Among strategies we considered were blocking on different size blocks, updating jointly with and without the hyperparameters, "oversampling" the parameters, hierarchical centering reparameterization, and "pilot adaptation" versus continuous tuning techniques for the proposal. A fixed Gaussian proposal even after a long tuning period does poorly; as the precision parameters change, a fixed proposal is unable to produce candidates that get accepted often enough for a good sample. Breaking the total area up into smaller blocks also does not appear to be a good solution since such schemes result in samplers with poorer mixing properties than algorithms which sample in one large block.

Knorr-Held and Rue (2002) suggest that blocking the hyperparameters with the model parameters is perhaps the only way to ensure good mixing for such models. Our schemes are easier to implement (they do not require use of specialized computer code), but at the same time can produce improvements in ESS *and*

	RUMCMC ESS	RSMCMC ESS	diff	ratio	RUMCMC ES/s	RSMCMC ES/s	diff	ratio
ϕ_{15}	25955	75468.5	49513	2.91	20.65	5.7	-14.95	0.28
θ_{17}	79213	94511.5	15298	1.19	63.02	7.14	-55.88	0.11
θ_{82}	93285	82025	-11260	0.88	74.21	6.19	-68.02	0.08

Table 4: Selected results for Minnesota cancer mortality data set. Each chain was run for 1 million iterations.



(a) RSMCMC ESS – RUMCMC ESS

(b) RSMCMC ES/s – RUMCMC ES/s

Figure 2 : Comparisons of ESS and ES per second, RUMCMC versus RSMCMC (positive values favor RSMCMC).

ES/s (the latter of which is not discussed by Knorr-Held and Rue). For the data sets considered here, our blocking scheme produces samples with good mixing even when the hyperparameters are sampled separately. In fact, sampling the τ_h and τ_c parameters separately and only at predetermined times resulted not only in much faster algorithms, but in posterior samples for the θ_is and ϕ_is having lower autocorrelations for all three data sets. Our experience with these different algorithms thus leads us to prefer the SMCMC and RSMCMC algorithms as a systematic way for efficient sampling from such models.

We described the results of applying our algorithm to three Minnesota cancer data sets. While SMCMC and RSMCMC provide better mixing in the samplers for several parameters in the first two data sets (breast and colorectal cancer detection), the univariate algorithm (UMCMC) produced samples that were adequate for most parameters though RUMCMC did rather poorly in comparison. However, with the third data set (cancer mortality), the same univariate sampler performed very poorly, often producing effective sample sizes of only 100 to 200 for 100,000 samples from the chain. The reparameterized univariate algorithm (RUMCMC) did perform much better, but the SMCMC and RSMCMC algorithms still outperformed it in terms of ESS, producing nearly uncorrelated samples, as well as reasonably high efficiency as measured by ES/s. Moreover, all of our results were based on only every tenth sample from the chain; our SMCMC algorithms would likely enjoy an even larger benefit had we not used this thinned sample.

A referee's suggestion led us to try our schemes on a simulated data set. We simulated θ_i and ϕ_i values from i.i.d. normal and CAR priors, respectively, given precision parameters $\tau_h = 250$ and $\tau_c = 100$, values roughly comparable to those observed in the posterior for the colorectal cancer data set. We then simulated data values Y_i from conditionally independent Poissons having mean $E_i \exp(\theta_i + \phi_i)$, where the E_is were also as given in the colorectal cancer data. We found that the SMCMC algorithms continued to outperform the univariate algorithms in much the same way as they did for the cancer detection and mortality data sets, with superior ESS and competitive ES/s results.

Overall, our experience with applying several SMCMC blocking schemes to real data sets suggests to us that SMCMC provides a standard, systematic technique for producing samplers with far superior mixing properties than simple univariate Metropolis-Hastings samplers. The SMCMC and RSMCMC schemes appear to be reliable ways of producing good ESSs, irrespective of the data sets and parameterizations. In many cases, the SMCMC algorithms are also competitive in terms of ES/s. On a more practical note, since our blocked SMCMC algorithms mix better, their convergence should be easier to diagnose and thus lead to final parameter estimates that are less biased. These estimates should also have smaller associated Monte Carlo variance estimates.

Results for the scheme using "full blocking" ($\boldsymbol{\Theta}$ and precision components) were generally promising in terms of fairly low autocorrelations for all the parameters. However, the ES/s values were never competitive, suggesting that for these data, much larger computation times are required. There are also several other techniques that could perhaps be used in conjunction with some of the SMCMC techniques described here to produce further improvements in the properties of the samples produced. For instance, multi-chain annealing or tempering (Geyer and Thompson, 1995; Neal, 1996) and simulated sintering (Liu and Sabatti, 1999) are recent approaches to accelerate sampling for such models. Also, there may be ways to further speed up computation by using linear algebra tricks for sparse matrices as described in Rue (2001) and Knorr-Held and Rue (2002).

References

[1] Bernardinelli, L., Clayton, D. and Montomoli, C. (1995) "Bayesian estimates of disease maps : How important are priors?" *Statistics in Medicine*, **14**, 2411-2431.

[2] Besag, J., Green, P., Higdon, D. and Mengerson, K. (1995) "Bayesian computation and stochastic systems," *Statistical Science*, **10**, 3–41.

[3] Besag, J. and Kooperberg, C. (1995) "On conditional and intrinsic autoregressions," *Biometrika*, **82**, 733–746.

[4] Besag, J., York, J. and Mollie, A. (1991) "Bayesian image restoration, with applications in spatial statistics (with discussion)," *Annals of the Institute of Statistical Mathematics*, **43**, 1–59.

[5] Best, N.G., Waller, L.A., Thomas, A., Conlon, E.M. and Arnold, R.A. (1999) "Bayesian models for spatially correlated diseases and exposure data," in *Bayesian Statistics 6*, J.M.Bernardo et al. , eds., Oxford: Oxford University Press, pp.131-156.

[6] Cancer Surveillance and Control Program (1997) "Case completeness and data quality audit: Minnesota Cancer Surveillance System 1994–1995," Technical report, Minnesota Department of Health.

[7] Clayton, D.G. and Kaldor, J. (1987) "Empirical Bayes estimates of age-standardized relative risks for use in disease mapping," *Biometrics*, **43**, 671-681.

[8] Cowles, M.K. (2002a) "Efficient model-fitting and model-comparison for high-dimensional Bayesian geostatistical models," to appear in *Journal of Statistical Planning and Inference.*

[9] Cowles, M.K. (2002b) "MCMC sampler convergence rates for hierarchical normal linear models : a simulation approach," to appear in *Statistics and Computing.*

[10] Eberly, L.E. and Carlin, B.P. (2000) "Identifiability and convergence issues for Markov chain Monte Carlo fitting of spatial models," *Statistics in Medicine*, **19**, 2279-2294.

[11] Gelfand, A., Sahu, S. and Carlin, B.P. (1995) "Efficient parameterizations for normal linear mixed models," *Biometrika*, **82**, 479-488.

[12] Geyer, C.J. (1992) "Practical Markov Chain Monte Carlo," *Statistical Science*, **7**, 473-483.

[13] Geyer, C.J., and Thompson, E.A. (1995) "Annealing Markov chain Monte Carlo with applications to ancestral inference," *Journal of the American Statistical Association*, **90**, 909-920.

[14] Haran, M. (2002) "Hierarchical Bayes modeling and computation for environmental exposure and health outcome assessment," Unpublished Ph.D. dissertation, School of Statistics and Division of Biostatistics, University of Minnesota.

[15] Hodges, J.S. (1998) "Some algebra and geometry for hierarchical models, applied to diagnostics (with discussion)," *Journal of the Royal Statistical Society, Ser. B*, **60**, 497-536.

[16] Hodges, J.S. and Carlin, B.P. (2001) "A note on the precision of the conditionally autoregressive prior in spatial models," Research Report 2001-024, Division of Biostatistics, University of Minnesota.

[17] Kass, R.E., Carlin, B.P., Gelman, A. and Neal, R. (1998) "Markov chain Monte Carlo in practice: a roundtable discussion," *American Statistician*, **52**, 93-100.

[18] Knorr-Held, L. and Rue, H. (2002) "On block updating in Markov random field models for disease mapping," to appear in *Scandinavian Journal of Statistics*.

[19] Liu, J.S. (1994) "The collapsed Gibbs sampler in Bayesian computations with applications to a gene regulation problem," *Journal of the American Statistical Association*, **89**, 958-966.

[20] Liu, J.S. and Sabatti, C. (1999) "Simulated sintering: Markov chain Monte Carlo with spaces of varying dimensions," in *Bayesian Statistics 6*, J.M.Bernardo et al. , eds., Oxford: Oxford University Press, pp.389-413.

[21] Liu, J.S., Wong, W.H. and Kong, A. (1994) "Covariance structure of the Gibbs sampler with applications to the comparisons of estimators and augmentation schemes," *Journal of the American Statistical Association*, **89**, 958-966.

[22] Neal, R.M. (1996). "Sampling from multimodal distributions using tempered transitions," *Statistics and Computing*, **6**, 353–366.

[23] Roberts G. and Sahu, S. (1997) "Updating schemes, correlation structure, blocking and parameterization for the Gibbs Sampler," *Journal of the Royal Statistical Society, Series B*, **59**, 291-317.

[24] Rue, H. (2001) "Fast sampling of Gaussian Markov Random Fields," *Journal of the Royal Statistical Society, Series B*, **63**, 325-338.

[25] Sargent, D.J., Hodges, J.S. and Carlin, B.P. (2000) "Structured Markov Chain Monte Carlo," *Journal of Computational and Graphical Statistics*, **9**, 217-234.

[26] Waller, L.A., Carlin, B.P., Xia, H., and Gelfand, A.E. (1997) "Hierarchical spatio-temporal mapping of disease rates," *Journal of the American Statistical Association*, **92**, 607–617.