

On Block Updating in Markov Random Field Models for Disease Mapping

LEONHARD KNORR-HELD

Lancaster University

HÅVARD RUE

Norwegian University of Science and Technology

ABSTRACT. Gaussian Markov random field (GMRF) models are commonly used to model spatial correlation in disease mapping applications. For Bayesian inference by MCMC, so far mainly single-site updating algorithms have been considered. However, convergence and mixing properties of such algorithms can be extremely poor due to strong dependencies of parameters in the posterior distribution. In this paper, we propose various block sampling algorithms in order to improve the MCMC performance. The methodology is rather general, allows for non-standard full conditionals, and can be applied in a modular fashion in a large number of different scenarios. For illustration we consider three different applications: two formulations for spatial modelling of a single disease (with and without additional unstructured parameters respectively), and one formulation for the joint analysis of two diseases. The results indicate that the largest benefits are obtained if parameters and the corresponding hyperparameter are updated jointly in one large block. Implementation of such block algorithms is relatively easy using methods for fast sampling of Gaussian Markov random fields (Rue, 2001). By comparison, Monte Carlo estimates based on single-site updating can be rather misleading, even for very long runs. Our results may have wider relevance for efficient MCMC simulation in hierarchical models with Markov random field components.

Key words: block updating, disease mapping, hierarchical models, Markov chain Monte Carlo, Markov random field models, shared component model

1. Introduction

There has been much recent interest in Bayesian hierarchical models in spatial epidemiology, as reviewed in Clayton & Bernardinelli (1992) and Wakefield *et al.* (2000). Such models provide a flexible way of handling spatial correlation in the data and can easily be combined with other models, such as models for measurement error of additional covariates (Bernardinelli *et al.*, 1997) or for temporal and spatio-temporal correlation (Waller *et al.*, 1997; Knorr-Held & Besag, 1998; Knorr-Held, 2000).

Many of the formulations used for spatial data are based on Gaussian Markov random field (GMRF) models. Statistical inference in such models is typically done by Markov chain Monte Carlo (MCMC) simulation methods, mostly by single-site updating, that is updating each parameter one by one in turn. However, it is well known from MCMC simulation in the related class of dynamic or state-space models that such single-site updating can have very poor convergence and mixing properties. Several authors have therefore suggested to block update the parameters in dynamic models (most of them using the Kalman filter/smoother), including Carter & Kohn (1994), Frühwirth-Schnatter (1994), Shephard & Pitt (1997) and Knorr-Held (1999), see also Wilkinson & Yeung (2002) who propose block updating in (nested hierarchical) linear models. An alternative way to improve mixing has been proposed by Gamerman (1998), who suggested reparametrizing the dynamic model to *a priori* independent system disturbances.

Similar problems with single-site updating occur in spatial models, where parameters are also correlated *a priori*. However, block updating has rarely been considered, because the lack of a (temporal) order in GMRF models makes it less obvious how to design fast algorithms for simulating from such a block at once. However, Rue (2001) has recently described a very efficient way of simulating from GMRF models, even if this involves a large number of parameters. Furthermore, he shows how to extend the method to non-Gaussian full conditional distributions, which involve GMRF terms. Such full conditionals typically arise when GMRF models are combined with non-normal observation models in a hierarchical framework, as in disease mapping, where Poisson or binomial observation models are combined with latent GMRF models for the unknown disease risk parameters.

In this paper we extend Rue's methodology in several ways and describe how block updating can be implemented in three typical disease mapping applications. For each application we test a whole range of possible blocking algorithms, including the two extreme cases single-site updating and updating of all or nearly all parameters in one block, on real data examples. Our blocking algorithms can roughly be categorized into two types: those which sample latent parameters jointly, but hyperparameters separately; and those which sample latent parameters and the corresponding hyperparameters in one block. Algorithms of the second type are novel and turn out to be very important for reliable MCMC simulation in such models.

In more complex hierarchical disease mapping models there are often additional unstructured parameters or even more than one GMRF component. We describe how to update all unknown parameters in one block, possibly even jointly with the relevant hyperparameters. This is possible in rather complicated settings with thousands of parameters. Our methods are also applicable if additional linear constraints are imposed on the GMRF components, a scenario where single-site updating is impractical due to degenerate full conditional distributions. We apply our algorithms exclusively in a disease mapping context; however, as we will note in the discussion, the proposed methodology can be used in many other areas of application.

Section 2 describes basic ideas underlying our algorithms in a generic fashion to indicate that the proposed methods have wider relevance in general hierarchical Bayesian models with GMRF components. This general framework includes so-called dynamic or state-space models for time-series or longitudinal data, for which block sampling algorithms based on the Kalman filter have been proposed in the literature (Carter & Kohn, 1994; Frühwirth-Schnatter, 1994). In appendix A we discuss the relationship between the algorithms based on the Kalman filter and Rue's (2001) Cholesky factor approach for Gaussian dynamic models. We conclude that in some situations the algorithms turn out to be essentially identical, but in general the Cholesky factor algorithm is conceptually simpler and computationally more efficient.

In section 3, we compare empirically a number of different block sampling algorithms for three specific spatial models. Implementation details can be found in appendix B. The first model, which is the simplest formulation, combines a Poisson likelihood with an intrinsic GMRF model for the unknown log relative risk parameters. The second model extends the first by adding additional unstructured parameters to the formulation and has been proposed in Besag *et al.* (1991). The third model we consider, a so-called shared component model, is built for a joint analysis of two diseases and includes three latent GMRF components, one which is shared by both diseases and two which are disease-specific, plus bivariate unstructured parameters for each district. Such a formulation has recently been proposed by Knorr-Held & Best (2000) and involves additional identifiability constraints on two of the three GMRF components. In all three models additional hyperparameters enter which are

treated as unknown and are assigned with suitable hyperpriors. Section 4 finally provides a discussion and outlines a number of other areas where block updating might prove beneficial.

2. Block simulation in GMRF models

This section starts with a review of the algorithm for simulating from a GMRF (Rue, 2001). We also sketch how Rue generates GMRF samples as an approximation to a non-standard full conditional distribution which involves a GMRF. This allows us to block update all parameters in more complex hierarchical models. In disease mapping applications the precision matrix of the GMRF typically depends on one or more additional hyperparameters, and we then describe a way to produce a joint sample of parameters and hyperparameters. Finally we discuss how these methods can be extended if additional linear constraints are imposed on the GMRF.

Let \mathbf{x} be a multivariate Gaussian random variable with regular precision matrix \mathbf{Q} and mean $\boldsymbol{\mu} = \mathbf{Q}^{-1}\mathbf{b}$, i.e.

$$\pi(\mathbf{x}) \propto \exp\left(-\frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{b}^T\mathbf{x}\right). \quad (1)$$

Such forms often arise for full conditional distributions in hierarchical models by combining the relevant product terms of the posterior distribution. The precision matrix \mathbf{Q} implies a conditional dependence structure for the components of \mathbf{x} with $Q_{ij} = 0$ if and only if x_i is conditionally independent of x_j , given all the other components of \mathbf{x} . If $Q_{ij} \neq 0$, then x_i is termed a neighbour of x_j . In disease mapping applications, the components of \mathbf{x} may correspond to district-specific risk parameters and x_i will be a neighbour of x_j if districts i and j share a common border. Of course, other definitions may be used as well.

Our methodology is not restricted to the adjacency graph defined by the contiguities of the n districts; if the hierarchical model consists of more parameters, then \mathbf{x} may contain more than n parameters. The adjacency graph is then a subgraph of a larger one defined by the conditional dependencies of the components of \mathbf{x} . For example, the model by Besag *et al.* (1991) involves n spatially structured and n additional unstructured parameters. For this model we will describe and implement a block update of all $2n$ parameters.

The algorithm by Rue (2001) proceeds in two steps. In the first step, the nodes of the graph are reordered so that the corresponding precision matrix has minimal bandwidth. For a given graph, this step has to be performed just once. For example, for the adjacency graph defined by the 366 districts in Sardinia, the bandwidth reduces from 244 to 36 after reordering. The precision matrix of the original and of the reordered graph can be seen in Fig. 1. Incidentally, Sardinia is divided into four provinces, which are easy to spot in the original precision matrix.

The reordered graph forms the basis for the application of a numerically efficient way of sampling from $\pi(\mathbf{x})$. The core of this ‘‘Cholesky factor algorithm’’ (CFA) is a numerically efficient Cholesky decomposition of the reordered precision matrix \mathbf{Q} into $\mathbf{L}\mathbf{L}^T$ which makes use of the band structure of \mathbf{Q} . Subsequently, n independent standard Gaussian random variables \mathbf{z} are generated and three systems of linear equations based on the Cholesky factor matrix \mathbf{L} and the vector \mathbf{b} are solved in order to produce the desired sample \mathbf{x} ; $\mathbf{x} = \boldsymbol{\mu} + \mathbf{u}$ where $\mathbf{L}^T\mathbf{u} = \mathbf{z}$, $\mathbf{L}\mathbf{v} = \mathbf{b}$ and $\mathbf{L}^T\boldsymbol{\mu} = \mathbf{v}$. This step can also be implemented in a numerically efficient way, as the (lower) bandwidth of \mathbf{L} is equal to the bandwidth of the reordered precision matrix \mathbf{Q} . There is a close connection between this approach and the Kalman-filter for dynamic models, which we discuss in appendix A.

In the first level of all models considered, disease counts are assumed to be conditionally independent Poisson distributed with mean equal to known expected counts times an unknown

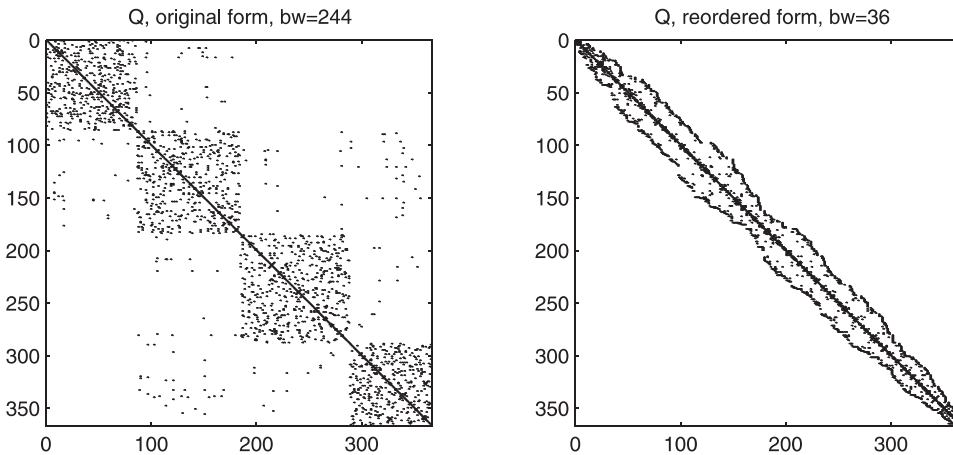


Fig. 1. Two precision matrices defined by the contiguities of the 366 districts of Sardinia. Non-zero elements are indicated by small dots. Left: Original graph. Right: Reordered graph.

relative risk. The log relative risk is then factorized into unknown parameters, so that the corresponding full conditional distributions are non-standard. For block updates of such non-standard full conditional distributions, we use a quadratic approximation to the non-Gaussian likelihood part of the full conditional, as described in Rue (2001). This allows us to use a GMRF sample as a proposal distribution in a Metropolis–Hastings step. The approximation could be based on a local Taylor expansion around the current value of \mathbf{x} . Alternatively, it could be chosen to provide a more global overall fit in order to yield higher acceptance probabilities. Even for a large number of parameters in the block (around 1000, say), the acceptance rates of such a Metropolis–Hastings proposal can well be around 20–50%.

Suppose now that the precision matrix \mathbf{Q} of the GMRF depends on additional hyperparameters $\boldsymbol{\theta}$. We generate a joint Metropolis–Hastings proposal $(\boldsymbol{\theta}, \mathbf{x})$ by first simulating from some (arbitrary) proposal distribution for $\boldsymbol{\theta}$, possibly depending on the current value of $\boldsymbol{\theta}$, but not depending on \mathbf{x} and subsequently sampling the GMRF $\pi(\mathbf{x}|\boldsymbol{\theta})$ as described above. The proposal $(\boldsymbol{\theta}, \mathbf{x})$ is then accepted or rejected jointly. Calculation of the acceptance probability will now involve the evaluation of the normalizing constant (which depends on \mathbf{Q} , hence on $\boldsymbol{\theta}$) of the density (1). This constant can be computed easily as a simple by-product of the sampling algorithm based on the diagonal elements of the Cholesky factor \mathbf{L} , as described in Rue (2001). Again, $\pi(\mathbf{x}|\boldsymbol{\theta})$ does not have to be a GMRF; in this case we compute a GMRF approximation to $\pi(\mathbf{x}|\boldsymbol{\theta})$ as above and use a sample from this approximation as a Metropolis–Hastings proposal.

In many applications \mathbf{Q} is a known structure matrix multiplied with an unknown scalar precision parameter θ (Clayton, 1996). Then we use a specific proposal for θ , multiplying the current value of θ with a random variable z proportional to $1 + 1/z$ on $[1/f, f]$, where $f > 1$ is a tuning constant. This proposal has the advantage that the proposal ratio in the Metropolis–Hastings acceptance probability equals one.

The proposed block updates are easily extended to the case where the GMRF is subject to a linear constraint $\mathbf{A}\mathbf{x} = \mathbf{c}$ by appropriate correction of the corresponding unconditional sample (Rue, 2001). For updates of \mathbf{x} without $\boldsymbol{\theta}$, computation of the prior density $\pi(\mathbf{x}|\mathbf{A}\mathbf{x})$ can be based on the identity $\pi(\mathbf{x}|\mathbf{A}\mathbf{x}) = \pi(\mathbf{A}\mathbf{x}|\mathbf{x})\pi(\mathbf{x})/\pi(\mathbf{A}\mathbf{x})$. As both $\pi(\mathbf{A}\mathbf{x}|\mathbf{x})$ and $\pi(\mathbf{A}\mathbf{x})$ are identical for the current and the proposed value of \mathbf{x} , the prior ratio in the Metropolis–Hastings acceptance probability reduces to the ratio of the unconstrained densities $\pi(\mathbf{x})$. However, for

joint updates of \mathbf{x} and additional hyperparameters $\boldsymbol{\theta}$, the normalizing constant of $\pi(\mathbf{Ax})$, which typically depends on $\boldsymbol{\theta}$, has to be taken into account as well.

3. Applications

3.1. Model 1

A common formulation to incorporate spatially structured heterogeneity in disease mapping is to assume that the observed disease counts y_i in district $i = 1, \dots, n$ are conditionally independent Poisson distributed with mean $e_i \exp(\eta_i)$, where e_i are known expected counts and η_i are the unknown log relative risk parameters, which are assumed to follow a GMRF. The most popular choice is a non-stationary “intrinsic autoregression”

$$\pi(\boldsymbol{\eta}|\kappa) \propto \kappa^{(n-1)/2} \exp\left(-\frac{\kappa}{2} \sum_{i \sim j} (\eta_i - \eta_j)^2\right) \quad (2)$$

although other formulations are possible, e.g. Cressie (1992). In (2), $i \sim j$ denotes all pairs of adjacent districts i and j . Formally, $\boldsymbol{\eta}$ follows a (singular) multivariate Gaussian distribution with “mean” zero and precision $\kappa \mathbf{K}$, where the i - j -off-diagonal element in the structure matrix \mathbf{K} is -1 if district i is adjacent to district j and zero elsewhere. The i th diagonal element in \mathbf{K} is equal to the number of neighbouring districts of district i . The formulation can be extended by additional weights, see Besag *et al.* (1991). This prior leaves the overall level of the GMRF unspecified, as only differences of log relative risk parameters enter in (2). An equivalent representation would be to include an additional intercept with an improper flat prior and a restriction imposed on the GMRF to have mean zero (Besag & Kooperberg, 1995).

Note that some authors use n instead of $n - 1$ for the degrees of freedom for κ in (2). We use $n - 1$ degrees of freedom because of the rank deficiency of \mathbf{K} with only $n - 1$ non-zero eigenvalues, see Knorr-Held (2002) for a detailed discussion. For the precision parameter κ we adopt the usual conjugate gamma prior $G(c, d)$, with density $\pi(\kappa) \propto \kappa^{c-1} \exp(-d\kappa)$, where c and d are suitably chosen constants.

We now report results of the algorithm for a dataset on insulin-dependent diabetes mellitus (IDDM) in Sardinia (Bernardinelli *et al.*, 1997). This is a sparse dataset with a total of 619 cases in $n = 366$ districts, where spatial smoothing is essential to get a realistic picture of the underlying risk surface. We set the hyperparameters to $c = 0.25$ and $d = 0.0005$, which has been suggested by Bernardinelli *et al.* (1995) as a vague prior choice.

We have implemented three sampling schemes: scheme 1 is a single-site algorithm; scheme 2 performs a block update of $\boldsymbol{\eta}$ and generates κ separately; and scheme 3 updates $\boldsymbol{\eta}$ and κ jointly (“hyperblock”). Details can be found in appendix B. Block updates in scheme 2 have acceptance rates of 74%. Here we have used a Taylor-approximation for the non-Gaussian terms in the full conditional distribution. In scheme 3 we have tuned the scaling parameter f of the proposal for κ so that the acceptance rates were slightly below 30%. Both scheme 2 and 3 did slightly more than 200 iterations per second on a DEC Alpha, whereas the single-site algorithm was approximately six times faster. For each of the three schemes we ran a chain of length 100 000 (including an initial burn-in period).

Figure 2 gives a plot of posterior samples (we have stored every 100th iteration) of the overall variation of the $\boldsymbol{\eta}$ parameters, measured as the logarithm of the sum of the squared differences $\sum_{i \sim j} (\eta_i - \eta_j)^2$, vs $\log \kappa$ for the different schemes. It can be seen that the two quantities are highly correlated which indicates that there are very strong dependencies between the log relative risk parameters $\boldsymbol{\eta}$ and the hyperparameter κ . Note that the posterior distribution for $\log \kappa$ is rather skewed with one long tail towards large values of κ . Any

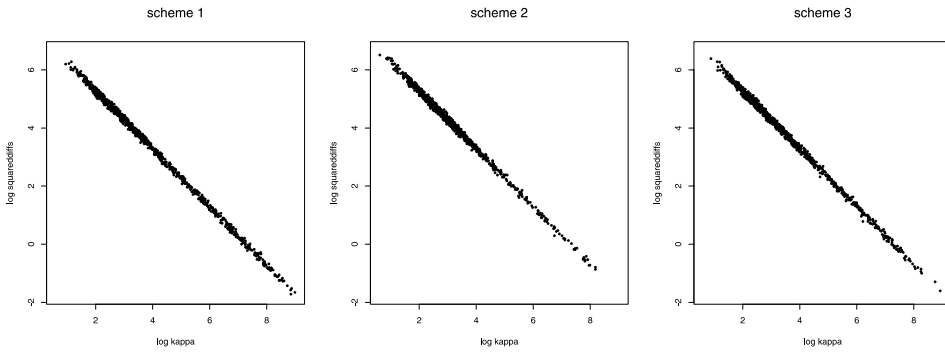


Fig. 2. Posterior samples of $\log(\sum_{i \sim j} (\eta_i - \eta_j)^2)$ vs $\log(\kappa)$.

MCMC algorithm with separate updates of κ and $\boldsymbol{\eta}$ may get stuck in this tail. Note that—for these runs—samples from scheme 1 seem to slightly overrepresent this tail, while in contrast scheme 2 does not fully exploit this tail. This can also be seen from Fig. 3, which gives trace-plots of $\log \kappa$ (again every 100th iteration, same runs) for the three different schemes. It is obvious from that figure that the mixing of κ for scheme 3 (with virtually independent samples) is much better than for the other two schemes.

For each district, we computed estimates (based on all 100 000 samples but ignoring those from the burn-in) of the posterior mean relative risk and the posterior probability of a relative risk above 1.0. These quantities are routinely calculated in disease mapping applications. In Fig. 4 these estimates (relative risk in the first column, posterior probabilities in the second) from scheme 1, 2, and 3 (first, second, and third row) are plotted against those obtained from a longer run (1 000 000 iterations) with scheme 3. The estimates from scheme 1 and 2 do not completely agree with those obtained from the longer run. Scheme 1 slightly underestimates the variation of the relative risk parameters because this run was oversampling the long tail of the posterior distribution of κ and high values of κ correspond to virtually no variation of the relative effect parameters. For scheme 2 an opposite effect can be seen, because this run was undersampling the tail of κ . One would not expect such discrepancies for MCMC estimates based on 100 000 iterations. The results suggest that only scheme 3 gives reliable estimates. We note that the runs presented here are not extreme cases but are typical for the amount of Monte Carlo error associated with the three different schemes.

3.2. Model 2

The model proposed in Besag *et al.* (1991) extends the formulation of section 3.1 by adding district-specific parameters accounting for additional unstructured heterogeneity. Rue (2001)

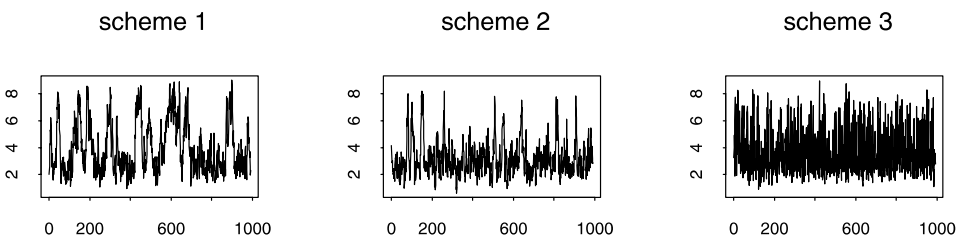


Fig. 3. Trace plots of $\log \kappa$ for the three different schemes.

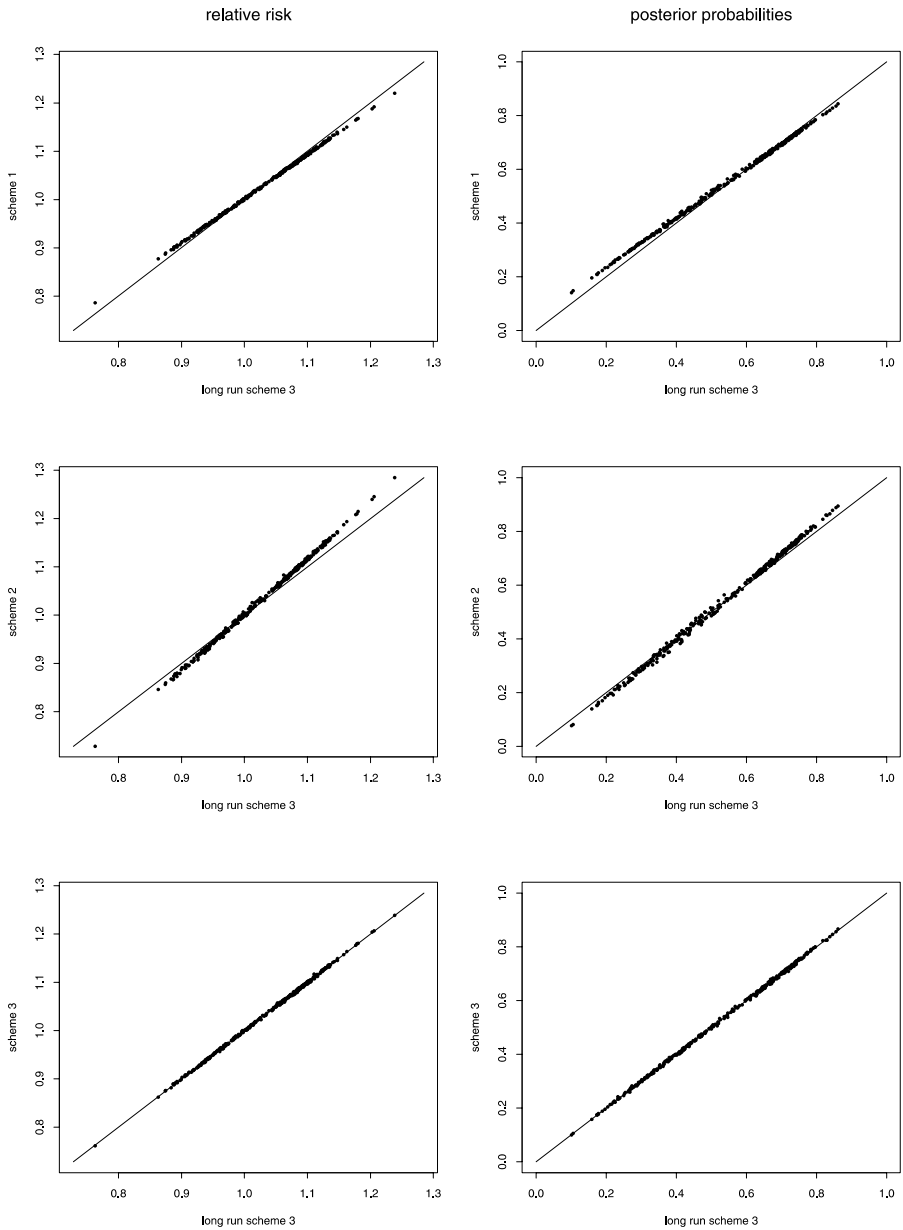


Fig. 4. Comparison of estimates based on 100 000 iterations with scheme 1, 2, and 3 with those obtained with a longer scheme 3 run. First column: relative risks. Second column: posterior probabilities.

has described a way to simulate from this model in the original parameterization by Besag *et al.* He uses a block update for the spatially structured parameters and single-site updates for all other parameters and hyperparameters. Here we follow Carlin & Louis (1996, p. 308) and reparametrize the model in order to facilitate the implementation of our blocking algorithms and to improve the performance of the MCMC algorithms. A similar approach has been suggested by Besag *et al.* (1995) for a Bayesian age–period–cohort model. One of the

advantages of the reparametrized model is that the full conditional for the spatially structured parameters is now multivariate Gaussian, so updating can be done with a simple (multivariate) Gibbs step. Also, in the original parameterization, there may be mixing problems with large (negative) posterior correlations between the spatial and non-spatial parameters for districts with a strong likelihood contribution. Finally, it is easier to design a block update of all (structured and unstructured) parameters in the reparametrized model, as the likelihood terms enter for only half of the parameters.

The reparametrized model can be written as a three-stage hierarchical model where, in the first-stage responses y_i are conditionally independent Poisson distributed with mean $e_i \exp(\eta_i)$, in the second-stage $\boldsymbol{\eta}$ is multivariate Gaussian with mean \mathbf{u} and diagonal precision matrix $\lambda \mathbf{I}$, and in the third-stage, \mathbf{u} follows a Markov random field with precision matrix $\kappa \mathbf{K}$. For λ and κ we adopt the usual (independent) gamma hyperpriors, say $\lambda \sim G(a, b)$ and $\kappa \sim G(c, d)$. We now report the results for the same Sardinia data as in model 1. We set $a = c = 0.25$, $b = 0.00025$ and $d = 0.0005$, which has been suggested by Bernardinelli *et al.* (1995) as a vague prior guess.

In total we have implemented 13 algorithms which differ in the way they form the blocks, and have also tested mixtures of those. To describe the different blocking schemes, we use the following notation. Let $[\boldsymbol{\eta}]$ denote an algorithm which updates $\boldsymbol{\eta}$ as a block and all other parameters by single-site. Similarly, $[\boldsymbol{\eta}, \lambda]$ denotes blocking of $\boldsymbol{\eta}$ and λ and $[\boldsymbol{\eta}, \lambda], [\mathbf{u}, \kappa]$ blocks $\boldsymbol{\eta}$ with λ and \mathbf{u} with κ . The different schemes are listed in Table 1. In the single-site algorithm, we use a log-gamma proposal for updating η_i , similar to model 1. For block updating $\boldsymbol{\eta}$ we use n such proposals and accept/reject them jointly. For block updates of u we can implement a Gibbs step as the full conditional is Gaussian with precision $\kappa \mathbf{K} + \lambda \mathbf{I}$ and mean $\lambda(\kappa \mathbf{K} + \lambda \mathbf{I})^{-1} \boldsymbol{\eta}$. For block updates of $\boldsymbol{\eta}$ and \mathbf{u} , we again use a Taylor-approximation for the non-Gaussian likelihood terms. We note here that the bandwidth of the precision matrix of $\boldsymbol{\eta}$ and \mathbf{u} reduces from 488 to 66 after reordering (not shown).

We also implemented two other algorithms: a bivariate block update of (η_i, u_i) , $i = 1, \dots, n$, and a Metropolis adjusted Langevin algorithm (e.g. Besag *et al.* (1995), who call it a Langevin-Hastings algorithm), which proposes to update all parameters in the direction of the gradient of the log-posterior. The first algorithm did not improve much over single-site, while for the latter the convergence was extremely slow. However, we didn't consider a modification of the

Table 1. Summary of the performance of the different blocking schemes. The categories are defined as "poor" (-), "moderate" (o) and "good" (+)

Index	Blocking scheme	Mixing			
		η	u	λ	κ
1	(single-site)	-	-	-	-
2	$[\mathbf{u}]$	-	-	-	-
3	$[\mathbf{u}, \kappa]$	-	-	-	-
4	$[\boldsymbol{\eta}]$	-	-	-	-
5	$[\boldsymbol{\eta}], [\mathbf{u}]$	-	-	-	-
6	$[\boldsymbol{\eta}], [\mathbf{u}, \kappa]$	-	-	-	-
7	$[\boldsymbol{\eta}, \lambda]$	-	-	+	-
8	$[\boldsymbol{\eta}, \lambda], [\mathbf{u}]$	-	-	+	-
9	$[\boldsymbol{\eta}, \lambda], [\mathbf{u}, \kappa]$	-	-	+	-
10	$[\boldsymbol{\eta}, \mathbf{u}]$	o	o	-	-
11	$[\boldsymbol{\eta}, \mathbf{u}, \lambda]$	o	o	+	-
12	$[\boldsymbol{\eta}, \mathbf{u}, \kappa]$	+	+	-	+
13	$[\boldsymbol{\eta}, \mathbf{u}, \lambda, \kappa]$	+	+	+	+

Metropolis adjusted Langevin algorithm based on a truncated proposal or a reparametrization of the model. Theoretical results suggest that this may improve the performance, although a referee has noted that despite good theoretical properties the Metropolis adjusted Langevin algorithm may mix slowly in practice if the parameters are very correlated.

For each scheme, we ran the algorithm for 100 000 iterations, following initial burn-in and tuning periods. Acceptance rates for blocks of $\boldsymbol{\eta}$ have been around 76%, and for blocks of \mathbf{u} and $\boldsymbol{\eta}$ around 70%. Figure 5 gives trace-plots of $\log \kappa$ and $\log \lambda$ for all 13 schemes. Here we have plotted every 100th iteration. As in model 1 we calculated and compared relative risk and posterior probability estimates (not shown).

Our main findings are indicated in Table 1, and can be summarized as follows:

1. All schemes which did not jointly update \mathbf{u} and κ produce rather unreliable relative risk estimates due to high posterior correlations between \mathbf{u} and κ ; similar to scheme 1 and 2 in model 1. Furthermore, even in scheme 3, 6 and 9 the parameters \mathbf{u} and κ still showed rather poor mixing with increased Monte Carlo simulation error and corresponding high variability of the relative risk estimates. Mixing of κ was satisfactory only in schemes 12 and 13, see Fig. 5.
2. Block updates of $\boldsymbol{\eta}$ without λ (schemes 4, 5, 6) did not improve over the corresponding single-site updates of $\boldsymbol{\eta}$ (schemes 1, 2, 3), see Fig. 5. This is not really surprising, as the η_i s are conditionally independent, given \mathbf{u} and λ . However, a joint update of $\boldsymbol{\eta}$ and λ (schemes 7, 8, 9) gives better mixing of λ , but not of the other parameters.
3. Joint updates of \mathbf{u} and $\boldsymbol{\eta}$ improved the mixing of these parameters. This indicates that there are high correlations between η_i and u_i , at least for some districts. Indeed, these correlations vary between 0.77 and 0.96 with a median correlation of 0.89. For this example, the spatially structured component dominates the estimated risk surface (posterior median of $\log \kappa$ equal to 3.3) with less variability of the spatially unstructured parameters, indicated by larger posterior values of the precision parameter λ (posterior median of $\log \lambda$ equal to 6.5).

This indicates that—for this dataset—reliable results can essentially only be achieved with block updates of all parameters (scheme 13). However, a problem with scheme 13 is that it is not immediately clear how to design and tune an appropriate proposal for κ and λ , before sampling $\mathbf{u}, \boldsymbol{\eta} | \kappa, \lambda$. We have used independent proposals with similar spread as those used in scheme 11 and 12 respectively. A promising alternative to scheme 13 is a mixture of schemes 11 and 12, which gives very similar results to scheme 13 and has the advantage that the spread of the proposal distribution for each hyperparameter can be tuned separately. The mixing is very similar to that of scheme 13, as can be seen in the last trace plot in Fig. 5. We finally note that for a more informative prior setting ($a = c = 1.0$, $b = 0.01$ and $d = 0.02$) the results are qualitatively similar with slightly better mixing of κ and λ .

3.3. Model 3

The third model we consider is a so-called shared component model and has recently been proposed (in a slightly different formulation) by Knorr-Held & Best (2000) for a joint spatial analysis of two diseases. The key idea is to separate the latent risk surfaces of the two diseases into three (spatially structured) components, one which is shared by both diseases, and two which are disease-specific. An additional scaling parameter δ allows for a different magnitude of the shared component for each of the two diseases. While Knorr-Held and Best use so-called cluster or partition models (Knorr-Held & Raßer, 2000) for the three spatial components, here we use GMRF models instead. Furthermore, we add bivariate Gaussian random variables to the formulation to account for unstructured heterogeneity.

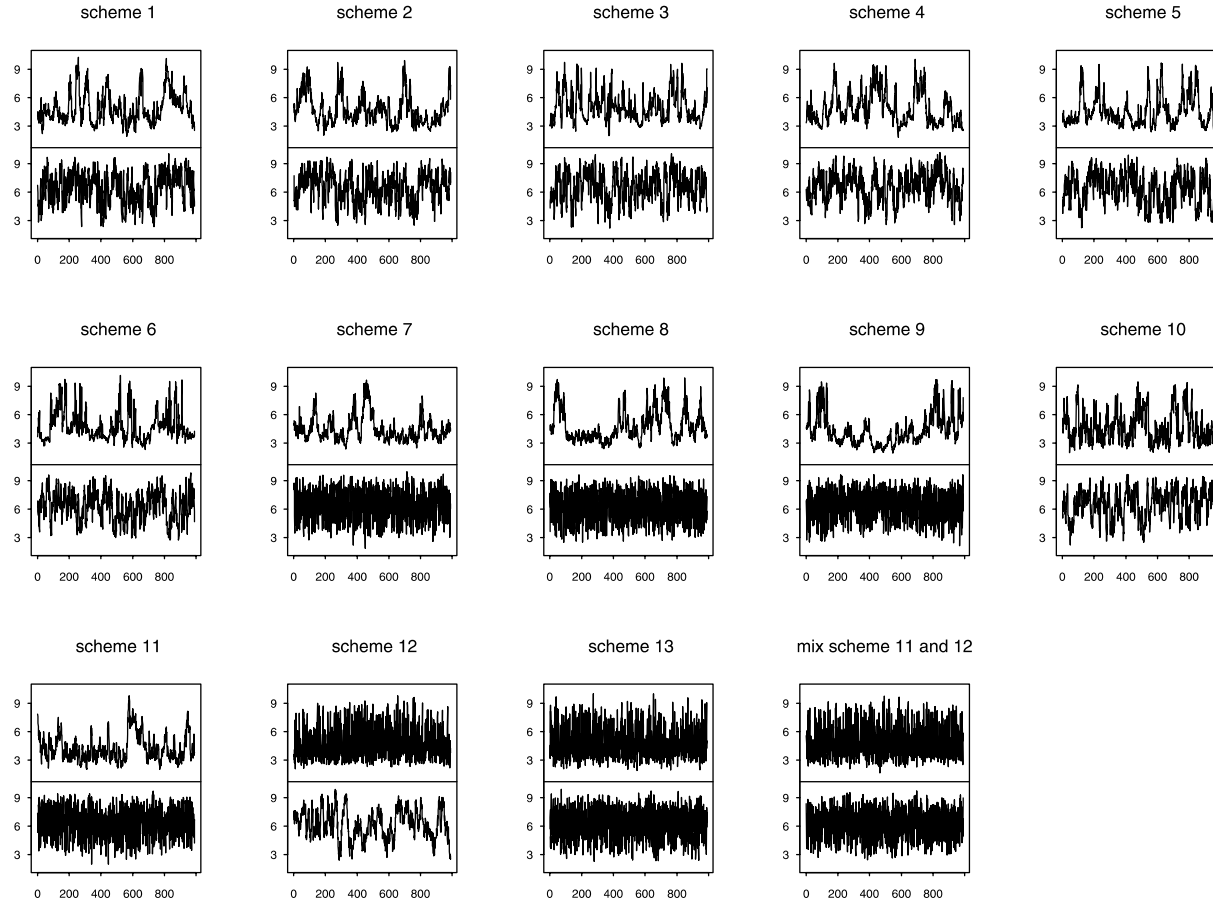


Fig. 5. Trace plots of $\log \kappa$ (upper plot) and $\log \lambda$ (lower plot) for schemes 1 to 13 and a mixture of scheme 11 and 12.

Let \mathbf{u} be the shared component and v_1 and v_2 the specific components for disease 1 and 2 respectively. Each of the three components is assumed to follow independently a GRMF (on the same graph) with precision parameters κ , v_1 and v_2 respectively. To ensure identifiability additional sum-to-zero restrictions $\sum_i v_{1i} = 0$ and $\sum_i v_{2i} = 0$ are imposed on the specific components.

Let $\eta_i = (\eta_{1i}, \eta_{2i})^T$ be the log relative risk parameter vector in district i for disease 1 and 2 respectively. We now assume that η_i is conditionally independent bivariate Gaussian with mean $(u_i \cdot \delta + v_{1i}, u_i/\delta + v_{2i})^T$ and precision matrix \mathbf{A} . Disease counts y_{di} , for disease $d = 1, 2$ in district i are assumed to be conditionally independent Poisson variables with mean $e_{di} \exp(\eta_{di})$.

For \mathbf{A} we adopt a Wishart prior with parameters a and \mathbf{B} , i.e. $\pi(\mathbf{A}) \propto |\mathbf{A}|^{a-3/2} \exp(-\text{tr}(\mathbf{B}\mathbf{A}))$, to allow for possible correlation between components of η_i . For κ , v_1 and v_2 we choose the usual gamma hyperpriors with parameters (c, d) , (e_1, f_1) and (e_2, f_2) , respectively. The logarithm of the scaling parameter δ is finally assumed to be normal with mean zero and variance τ^2 .

Again, we have implemented a large number of different block updating algorithms to sample from the implied posterior distribution. For joint updates of parameters, possibly with hyperparameters, we have written a generic update routine which can be called by specifying which of the parameters with which of the hyperparameters one wants to block. More details can be found in the user-manual in the library GMRFsim, see www.math.ntnu.no/~hrue/GMRFsim. Joint updates of parameters and hyperparameters always first propose a new value for the hyperparameter as in model 1 and 2, and then sample the parameter block, given the proposed new hyperparameter values. Note that a complete single-site algorithm is in fact impossible, as the full conditional of any component v_{di} will have point mass one at the current value, to ensure the sum-to-zero restriction. We have therefore always used block updates for the disease-specific GMRFs v_1 and v_2 .

We briefly report here only results from three different block updating schemes. Scheme 1 updates \mathbf{u} jointly with κ , v_1 jointly with v_1 , and v_2 jointly with v_2 . We have tuned each of the proposals for the hyperparameters to achieve acceptance rates of 25–30% for each of the three blocks. Scheme 2 updates \mathbf{u} , v_1 and v_2 in one block jointly with one of the hyperparameters δ , κ , v_1 and v_2 in turn. This allows us again to tune each of the four different block updates to have acceptance rates around 25–30% (Note that the full conditional for $[\mathbf{u}, v_1, v_2]$ is multivariate Gaussian, hence updating of this block without a hyperparameter can be done with a Gibbs step). In both schemes we use log-gamma proposals to update the η_{di} s by single-site, similar as in model 1 and 2.

The third scheme consists of block updates of all $5 \times 544 = 2720$ parameters \mathbf{u} , v_1 , v_2 , $\boldsymbol{\eta}_1$ and $\boldsymbol{\eta}_2$, again jointly with one of the hyperparameters in turn. Figure 6 gives the corresponding precision matrix of the GMRF on all these parameters before and after reordering. The original order of the parameters is $u_1, v_{11}, v_{21}, \eta_{11}, \eta_{21}, u_2, v_{12}, v_{22}, \eta_{11}, \eta_{22}, \dots$. Acceptance rates for a block update $[\mathbf{u}, v_1, v_2, \boldsymbol{\eta}_1, \boldsymbol{\eta}_2]$ without a hyperparameter are still acceptable with values above 16%. Here we have used an approximation to the likelihood based on Rue’s (2001, top of p. 335) proposal, which approximately doubles the acceptance rates compared to the conceptually simpler Taylor-approximation. We have tuned the joint updates with each of the four hyperparameters to have acceptance rates around 10–13%. We note here that, in principle, we could easily construct a move that updates all parameters together with more than one or even all hyperparameters, but the choice of the scaling parameters of the proposals for the hyperparameters is not obvious. Therefore, in the spirit of mixing scheme 11 and 12 in model 2, we update only one of the hyperparameters in turn, together with the whole parameter block.

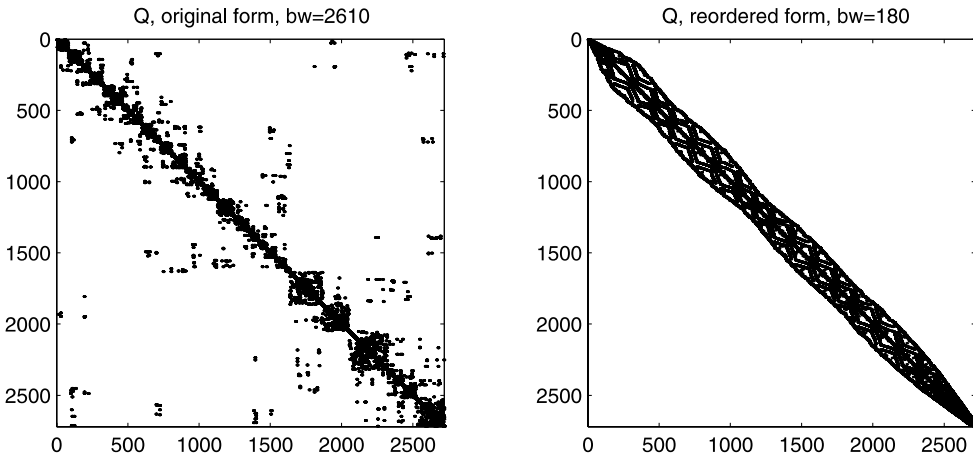


Fig. 6. The two precision matrices defined by the dependencies of the 2720 parameters in model 3. Non-zero elements are indicated by small dots. Left: original graph. Right: reordered graph.

We finally note a technicality. For sampling from a GMRF under a linear constraint we usually sample from the unconstrained version and correct the sample as described in Rue (2001). This is the way we have updated v_1 and v_2 in scheme 1. However, this requires the unconstrained GMRF to be proper. For joint updates of \mathbf{u} , v_1 and v_2 (and possibly η_1 and η_2), however, the unconstrained GMRF is improper due to the three implicit flat priors on the overall level. We have therefore a small value ϵ on the diagonal of the precision matrix to make it proper, and have corrected a sample from this unconstrained GMRF as a Metropolis–Hastings proposal. In the acceptance step, we then adjust for the modified proposal distribution, so our algorithm remains valid.

We applied the different schemes to data on oral cavity and oesophageal cancer mortality in the 544 districts of Germany, 1986–1990, already analysed in Knorr-Held & Best (2000). These data are not particularly sparse so we do not expect such severe problems with single-site algorithms as for the Sardinia data. Nevertheless, given the large number of parameters in the model, we still hope to see improvements in mixing for the block algorithms. For the analysis we have set $a = 1.5$, $\mathbf{B} = \text{diag}(0.01)$, $c = e_1 = e_2 = 1.0$, $d = f_1 = f_2 = 0.02$ and $\tau^2 = 0.17$.

For each scheme, we ran a chain of 100 000 iterations, storing every 100th sample. The different algorithms did approximately five iterations per second for scheme 1, three for scheme 2 and two for scheme 3. Despite the lower acceptance rates of scheme 3, mixing of the hyperparameters was even slightly better than for scheme 1 and 2 (not displayed). However, a simpler scheme with block updates of v_1 and v_2 and single-site updates of all other parameters performed equally well. Nevertheless, this example illustrates that we can apply our block algorithm to such a problem, if needed. Based on the empirical findings from model 1 and 2, we expect to see more improvements of the blocking algorithm for sparser data.

4. Discussion

This paper has demonstrated the use of block updating algorithms in Bayesian hierarchical models for disease mapping. In the first two models considered, we have shown that joint updates of GMRF parameters together with hyperparameters may be necessary to get reliable estimates of relative risk parameters or related quantities. Such joint updates ensure better

mixing and hence induce smaller simulation error for parameter estimates. They prevent the chain from getting trapped in long tails of the posterior distribution, which possibly leads to unreliable estimates; even for very long runs. These advantages seem to very much compensate for the additional cost in computing time and coding.

For model 3 we have been able to design block update algorithms on a large graph, induced by a complicated hierarchical model built upon three latent GMRFs plus exchangeable parameters and various hyperparameters. This was done to illustrate that the proposed methodology is generic and can be applied in rather different scenarios. Also it was shown that the methods allow for proper incorporation of identifiability restrictions, which would be not possible for any single-site algorithm.

Of course, there is always a limit for any blocking algorithm of non-standard full conditionals, if the number of parameters considered gets very large. However, in disease mapping algorithms, the number of districts rarely exceeds a few thousand, and for such problems our algorithms seem to work fine.

There is a wide range of applications of the proposed methodology outside of disease mapping applications. For example, Fahrmeir & Lang (2001) recently described several formulations based on Markov random field priors for Bayesian non- and semi-parametric inference in generalized additive models, see also Hastie & Tibshirani (2000). Other applications are models for space-time interactions based on GMRF priors (Clayton, 1996, Knorr-Held, 2000), models for agricultural field experiments (Besag & Higdon, 1999) or Bayesian versions of age-period-cohort models (Besag *et al.*, 1995). We note here that the block algorithms are not restricted to Gaussian MRFs; the adoption of scale mixtures of normals allows for many other distributional assumptions, including the popular *t*-distributions, possibly even with an unknown number of degrees of freedom, see Besag *et al.* (1995) and Besag & Higdon (1999). A GMRF with a sparse precision matrix may also be used to approximate a (stationary) Gaussian field, specified through a given correlation structure (Rue & Tjelmeland, 2002). In geostatistical applications, the correlation matrix of the Gaussian field might depend on a few unknown hyperparameters, and it would be interesting to study if our proposed joint updates of the GMRF approximation together with these hyperparameters are applicable as well.

Acknowledgements

We acknowledge support from the German Science Foundation (DFG), SFB 386, the EU TMR network ERB-FMRX-CT96-0095 on "Computational and statistical methods for the analysis of spatial data", and from the European Science Foundation Programme on Highly Structured Stochastic Systems (HSSS). We thank Luisa Bernardinelli for providing the dataset on IDDM. The revision has benefited from comments by the associate editor and two referees.

References

- Bernardinelli, L., Clayton, D. & Montomoli, C. (1995). Bayesian estimates of disease maps: how important are priors? *Statist. Med.* **14**, 2411–2431.
- Bernardinelli, L., Pascutto, C., Best, N. G. & Gilks, W. R. (1997). Disease mapping with errors in covariates. *Statist. Med.* **16**, 741–752.
- Besag, J. E. & Higdon, D. (1999). Bayesian analysis of agricultural field experiments (with discussion). *J. Roy. Statist. Soc. Ser. B* **61**, 691–746.
- Besag, J. E. & Kooperberg, C. (1995). On conditional and intrinsic autoregressions. *Biometrika* **82**, 733–746.

- Besag, J. E., York, J. C. & Mollié, A. (1991). Bayesian image restoration with two applications in spatial statistics (with discussion). *Ann. Inst. Statist. Math.* **43**, 1–59.
- Besag, J. E., Green, P. J., Higdon, D. M. & Mengersen, K. L. (1995). Bayesian computation and stochastic systems (with discussion). *Statist. Sci.* **10**, 3–66.
- Carlin, B. P. & Louis, T. A. (1996). *Bayes and empirical Bayes methods for data analysis*. Chapman & Hall, London.
- Carter, C. K. & Kohn, R. (1994). On Gibbs sampling for state space models. *Biometrika* **81**, 541–553.
- Clayton, D. G. (1996). Generalized linear mixed models. In: *Markov chain Monte Carlo in practice* (eds W. R. Gilks, S. Richardson & D. J. Spiegelhalter), 275–301. Chapman & Hall, London.
- Clayton, D. G. & Bernardinelli, L. (1992). Bayesian methods for mapping disease risks. In: *Small area studies in geographical and environmental epidemiology* (eds J. Cuzick & P. Elliot), 205–220. Oxford University Press, Oxford.
- Cressie, N. (1992). Smoothing regional maps using empirical Bayes predictors. *Geographical Anal.* **24**, 75–95.
- De Jong, P. & Shephard, N. (1995). The simulation smoother for time series models. *Biometrika* **82**, 339–350.
- Fahrmeir, L. & Kaufmann, H. (1991). On Kalman filtering, posterior mode estimation and Fisher scoring in dynamic exponential family regression. *Metrika* **38**, 37–60.
- Fahrmeir, L. & Lang, S. (2001). Bayesian inference for generalized additive mixed models based on Markov random field priors. *J. Roy. Statist. Soc. Ser. C* **50**, 201–220.
- Frühwirth-Schnatter, S. (1994). Data augmentation and dynamic linear models. *J. Time Ser. Anal.* **15**, 183–202.
- Gamerman, D. (1998). Markov chain Monte Carlo for dynamic generalized linear models. *Biometrika* **85**, 215–227.
- Hastie, T. & Tibshirani, R. (2000). Bayesian backfitting (with discussion). *Statist. Sci.* **15**, 196–223.
- Knorr-Held, L. (1999). Conditional prior proposals in dynamic models. *Scand. J. Statist.* **26**, 129–144.
- Knorr-Held, L. (2000). Bayesian modeling of inseparable space-time variation in disease risk. *Statist. Med.* **19**, 2555–2567.
- Knorr-Held, L. (2002). Some remarks on Gaussian Markov random field models for disease mapping. In: *Highly structured stochastic systems* (eds N. Hjort, P. Green & S. Richardson), Oxford University Press, Oxford, to appear. Available at www.stat.uni-muenchen.de/~leo/publikationen.html.
- Knorr-Held, L. & Besag, J. (1998). Modelling risk from a disease in time and space. *Statist. Med.* **17**, 2045–2060.
- Knorr-Held, L. & Best, N. G. (2000). A shared component model for detecting joint and selective clustering of two diseases. *J. Roy. Statist. Soc. Ser. A* **164**, 73–85.
- Knorr-Held, L. & Raßer, G. (2000). Bayesian detection of clusters and discontinuities in disease maps. *Biometrics* **56**, 13–21.
- Künsch, H.-R. (2001). State space and hidden Markov models. In *Complex stochastic systems* (eds O. E. Barndorff-Nielsen, D. R. Cox & C. Klüppelberg), pp. 109–173. Chapman & Hall/CRC, Boca Raton.
- Rue, H. (2001). Fast sampling of Gaussian Markov random fields. *J. Roy. Statist. Soc. Ser. B* **63**, 325–338.
- Rue, H. & Tjelmeland, H. (2002). Fitting Gaussian Markov random fields to Gaussian fields. *Scand. J. Statist.* **29**, 31–49.
- Shephard, N. & Pitt, M. K. (1997). Likelihood analysis of non-Gaussian measurement time series. *Biometrika* **84**, 653–667.
- Wakefield, J. C., Best, N. G. & Waller, L. A. (2000). Bayesian approaches to disease mapping. In *Spatial epidemiology: methods and applications* (eds P. Elliot, J. C. Wakefield, N. G. Best & D. J. Briggs), 104–127. Oxford University Press, Oxford.
- Waller, L. A., Carlin, B. P., Xia, H. & Gelfand, A. E. (1997). Hierarchical spatio-temporal mapping of disease rates. *J. Amer. Statist. Assoc.* **92**, 607–617.
- Wilkinson, D. J. & Yeung, S. K. H. (2002). Conditional simulation from highly structured Gaussian systems, with application to blocking-MCMC for the Bayesian analysis of very large linear models. *Statist. Comput.* **12**.

Received September 2000, in final form September 2001

Leonhard Knorr-Held, Department of Mathematics and Statistics, Fylde College, Lancaster University, Lancaster LA1 4YF, UK.
E-mail: l.knorr-held@lancaster.ac.uk

Appendix A. The relation between the Cholesky factor algorithm and the Kalman-filter

Consider the Gaussian dynamic or state-space model, $t = 1, \dots, n$

$$\mathbf{x}_t \mid \text{past} \sim \mathcal{N}(\mathbf{G}_t \mathbf{x}_{t-1}, \boldsymbol{\Sigma}_t) \tag{3}$$

$$\mathbf{y}_t \mid \mathbf{x}_t, \text{past} \sim \mathcal{N}(\mathbf{H}_t \mathbf{x}_t, \boldsymbol{\Omega}_t), \tag{4}$$

where \mathbf{x}_t is the (hidden) Gaussian Markov chain with k -dimensional states, \mathbf{G}_t is a $k \times k$ -matrix, \mathbf{H}_t is a $l \times k$ matrix, $\boldsymbol{\Sigma}_t$ is a k -dimensional covariance matrix, and \mathbf{y}_t are l -dimensional Gaussian observations with mean $\mathbf{H}_t \mathbf{x}_t$ and covariance $\boldsymbol{\Omega}_t$. It is well known that we can use the Kalman-filter to:

- (1) sample exactly from $\pi(\mathbf{x}_1, \dots, \mathbf{x}_n \mid \mathbf{y}_1, \dots, \mathbf{y}_n)$, and
- (2) compute the normalization constant for the same conditional density,

in terms of $\mathcal{O}(n)$ flops for fixed k and l . This sampling algorithm is usually called the forward-filtering-backward-sampling (FFBS) algorithm. This appendix discusses the relation between the FFBS algorithm (and its more recent variants) and the Cholesky factor algorithm (CFA) used in this study, for sampling from the GMRF defined in (1). There is a close correspondence between the two algorithms, as the Kalman-filter/smoothener relates nicely to entries in the Cholesky factor of the precision matrix (Fahrmeir & Kaufmann, 1991).

A.1. The forward-filtering-backward-sampling algorithm

To simplify the notation, let $\mathbf{y}_1^n = (\mathbf{y}_1, \dots, \mathbf{y}_n)$ and define similarly \mathbf{x}_1^n . A sample from $\mathbf{x}_1^n \mid \mathbf{y}_1^n$ can be generated by first doing a forward-filtering (FF) step using the Kalman-filter and then a backward-sampling (BS) step (Carter & Kohn, 1994, Frühwirth-Schnatter, 1994). Similarly, the normalized likelihood can be evaluated for any fixed state. The algorithm proceeds in two steps: In the FF-step, we compute the filtering densities sequentially from $t = 1$ to n , by

$$\pi(\mathbf{x}_t \mid \mathbf{y}_1^t) \propto \int_{\mathbf{x}_{t-1}} \pi(\mathbf{x}_{t-1} \mid \mathbf{y}_1^{t-1}) \pi(\mathbf{x}_t \mid \mathbf{x}_{t-1}) \pi(\mathbf{y}_t \mid \mathbf{x}_t) d\mathbf{x}_{t-1}.$$

The integration is trivial to do analytically since all involved densities are Gaussian. We have now access to $\pi(\mathbf{x}_n \mid \mathbf{y}_1^n)$ which is the starting-point for the BS-step which goes backward in time. The conditional independence structure of the model implies that

$$\pi(\mathbf{x}_{n-1} \mid \mathbf{x}_n, \mathbf{y}_1^n) = \pi(\mathbf{x}_{n-1} \mid \mathbf{x}_n, \mathbf{y}_1^{n-1}) \propto \pi(\mathbf{x}_{n-1} \mid \mathbf{y}_1^{n-1}) \pi(\mathbf{x}_n \mid \mathbf{x}_{n-1}). \tag{5}$$

Note that $\pi(\mathbf{x}_{n-1} \mid \mathbf{y}_1^{n-1})$ is already computed in the FF-step, so (5) is easy to compute. We continue the process backward in time using the obvious generalization of (5) for general $t = n - 1, \dots, 1$, obtaining

$$\pi(\mathbf{x}_1^n \mid \mathbf{y}_1^n) = \pi(\mathbf{x}_n \mid \mathbf{y}_1^n) \prod_{t=n-1}^1 \pi(\mathbf{x}_t \mid \mathbf{x}_{t+1}^n, \mathbf{y}_1^n). \tag{6}$$

$$= \pi(\mathbf{x}_n \mid \mathbf{y}_1^n) \prod_{t=n-1}^1 \pi(\mathbf{x}_t \mid \mathbf{x}_{t+1}^t, \mathbf{y}_1^t). \tag{7}$$

Note that (7) is sequential backward in time, hence \mathbf{x}_1^n can be sampled by first sampling \mathbf{x}_n , then \mathbf{x}_{n-1} conditional on \mathbf{x}_n and so on. Further, we have access to the normalized joint density, as the normalization constant is just a product of n normalization constants of k -dimensional Gaussian distributions.

A.2. The Cholesky factor algorithm

We now apply the Cholesky factor algorithm (CFA) of Rue (2001) to the Gaussian dynamic model defined in (3) and (4). No reordering of the vertices is needed in this case. To simplify the discussion we assume Σ_t is non-singular for all t . This assumption will be relaxed later on. Note that $\pi(\mathbf{x}_1^n | \mathbf{y}_1^n)$ is Gaussian (as defined in (1)) with block-tridiagonal precision matrix \mathbf{Q} (with k -dimensional blocks) and $\mathbf{b}(= \mathbf{b}_1^n)$ containing the contribution from the observed data. The specific elements of \mathbf{Q} and \mathbf{b} are not needed for the following, so we don't give more details. Denote by \mathbf{L} the block-triangular (with k -dimensional blocks) Cholesky factor of \mathbf{Q} and let the ij th block of \mathbf{L} be \mathbf{L}_{ij} .

A sample from $\pi(\mathbf{x}_1^n | \mathbf{y}_1^n)$ can now be generated (compare section 2) by $\mathbf{x}_1^n = \mathbf{u}_1^n + \boldsymbol{\mu}_1^n$ where \mathbf{u}_1^n is the solution of $\mathbf{L}^T \mathbf{u}_1^n = \mathbf{z}_1^n$ where the \mathbf{z}_t s are independent k -dimensional Gaussian with zero mean and covariance \mathbf{I} , and the mean $\boldsymbol{\mu}_1^n$ is the solution of $\mathbf{L} \mathbf{v}_1^n = \mathbf{b}_1^n$ and $\mathbf{L}^T \boldsymbol{\mu}_1^n = \mathbf{v}_1^n$. By writing out the equations (the derivation is based on standard matrix-algebra) we finally obtain (with obvious changes when $t = n$)

$$\text{Prec}(\mathbf{u}_t | \mathbf{u}_{t+1}^n, \mathbf{y}_1^n) = \mathbf{L}_{tt} \mathbf{L}_{tt}^T \tag{8}$$

$$E(\mathbf{u}_t | \mathbf{u}_{t+1}^n, \mathbf{y}_1^n) = -\mathbf{L}_{tt}^{-T} \mathbf{L}_{t,t-1}^T \mathbf{L}_{t,t-1} \mathbf{u}_{t+1} \tag{9}$$

$$\boldsymbol{\mu}_t = -\mathbf{L}_{tt}^{-T} (\mathbf{v}_t - \mathbf{L}_{t,t-1}^T \boldsymbol{\mu}_{t+1}); \tag{10}$$

here Prec denotes the precision matrix. Hence, the diagonal terms \mathbf{L}_{tt} of \mathbf{L} are the Cholesky factors of the conditional precisions and the off-diagonal terms $\mathbf{L}_{t,t-1}$ are related to the conditional expectations.

Note that (8), (9) and (10) simply compute all terms needed in (6), and provide formula (7) itself. (The simplification from (6) to (7) is however not immediate from (8), (9) and (10), but we omit the detailed argument here for simplicity.) Hence, the FFBS algorithm using the Kalman-filter is equivalent to the CFA, in the meaning that they compute the same conditional densities needed in (6). The minor differences are that the CFA computes the Cholesky factor of the precision matrix for each t (see (8)), while the Kalman-filter usually computes the corresponding covariance matrix. The Kalman-filter also computes the mean and covariance of $\mathbf{x}_t | \mathbf{y}_1^t$ for each t in the FF-step, which introduce minor redundancy since only the conditional densities in (7) are needed. The CFA computes directly the conditional densities and nothing else.

The equivalence can also be used to derive the Kalman-recursions directly from the sequence of precision matrices for $\mathbf{x}_1^t | \mathbf{y}_1^t$, $t = 1, \dots, n$ and their Cholesky factors, but we omit this detail here.

A.3. When do they differ?

The difference between the FFBS algorithm and the CFA becomes clear when Σ_t is singular or we simply do not have a dynamic model to start with.

Singularity of Σ_t is typically encountered when forcing a model into the state space form (3). As an illustration, consider a standard auto-regressive process of order p ,

$$x_t - \phi_1 x_{t-1} - \dots - \phi_p x_{t-p} = \epsilon_t \tag{11}$$

where $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$ and with observations $y_t \sim \mathcal{N}(x_t, \tau^2)$. To use the Kalman-filter to sample $\mathbf{x}_1^n | \mathbf{y}_1^n$, we put (11) into a state-space form with $k = p$ and rank of $\Sigma_t = 1$ (to account for the deterministic relations). Using the state-space representation the precision matrix of $\mathbf{x}_1^n | \mathbf{y}_1^n$ has dimension nk but only rank n . It is easy and theoretically convenient to

use deterministic relations to force models into the state-space form, but it is quite hard to account for this in an algorithmical implementation of the Kalman-filter. Further, it also slows down the computation. Frühwirth-Schnatter (1994, sect. 3) and DE Jong & Shephard (1995) suggested applying the Kalman-filter only for the non-deterministic part of (3), hence reducing the dimension of the model from nk to n . Using the CFA, we do not encounter this problem at all as the precision matrix for $\mathbf{x}_1^n | \mathbf{y}_1^n$ is a band-matrix with bandwidth p , hence \mathbf{L} is lower triangular with the same bandwidth (Rue, 2001). The CFA does not rely on a dynamical representation like (3), but only on the band-structure of the precision-matrix.

Although dynamic models are important, we are most interested in spatial applications, where the Kalman-filter does not apply: The prior model (1) is defined jointly and there is no “time” nor an easy-to-get forward representation like (3). The CFA however can still be used, possibly after reordering of the indices to obtain a small bandwidth in order to speed up the computation. The CFA will still provide us with a representation like (7) defined backward in “time”, with the correct joint density.

The CFA seems to be superior to the Kalman-filter, as it offers great simplification conceptually, the same computer-code can be used for Markov models in time and in space (or even in space-time), both on lattices and graphs, and the algorithm can make use of efficient algorithms for computing the (band) Cholesky factorization and solving (band) linear systems. Although the Kalman-filter can be coded using the same linear algebra software, the efficiency will typically be less as the implementation will involve more calculations applied on $k \times k$ matrices repeated n times, instead of having the critical calculation done on one large (band-)matrix of dimension nk .

Nevertheless, the FFBS algorithm is still very important, as it is valid not only in the Gaussian case but for any model with the same conditional independence structure as the state-space model. For example, the sequence of unknown states in a hidden Markov model (for a recent review see Künsch, 2001) can be generated jointly with the FFBS algorithm.

Appendix B. Implementation details

Due to space limitations we give details only for Model 1, the block update schemes for Model 2 and 3 are based on the same ideas. For single-site updating (scheme 1), we sample κ from its full conditional distribution $G(c + (n - 1)/2, d + \sum_{i \sim j} (\eta_i - \eta_j)^2/2)$ while for updating η_i , we use a log-gamma Metropolis–Hastings proposal as an approximation to the non-standard full conditional (acceptance rates around 99%). More specifically, we use the logarithm of a $G(y_i + \mu_i^2/\sigma_i^2, e_i + \mu_i/\sigma_i^2)$ random variable where μ_i and σ_i^2 are the mean and variance of the conditional (lognormal) distribution of $\exp(\eta_i) | \boldsymbol{\eta}_{j \neq i}, \kappa$. These parameters are simple functions of the parameters of the conditional (normal) distribution of $\eta_i | \boldsymbol{\eta}_{j \neq i}, \kappa$. For the intrinsic autoregression (2), $\mu_i = \exp(\bar{\eta}_i + 1/(2n_i\kappa))$ and $\sigma_i^2 = \exp(2\bar{\eta}_i + 1/(n_i\kappa))(\exp(1/(n_i\kappa)) - 1)$, where $\bar{\eta}_i$ is the corresponding mean value over the n_i districts that are geographically contiguous to i .

To construct the block update in scheme 2, we start with the full conditional for $\boldsymbol{\eta}$,

$$\pi(\boldsymbol{\eta} | \kappa, \mathbf{y}) \propto \exp\left(-\frac{\kappa}{2} \sum_{i \sim j} (\eta_i - \eta_j)^2 + \sum_i y_i \eta_i - \sum_i e_i \exp(\eta_i)\right). \tag{12}$$

We use a GMRF approximation to (12) as a proposal distribution in a Metropolis–Hastings step. We replace the term $\exp(\eta_i)$ by a quadratic approximation around a suitable point η_i^* , for example by using Taylor expansion

$$\begin{aligned} \exp(\eta_i) &\approx \exp(\eta_i^*) \left(1 + (\eta_i - \eta_i^*) + \frac{1}{2} (\eta_i - \eta_i^*)^2 \right) \\ &= c_i(\eta_i^*)\eta_i + \frac{1}{2} d_i(\eta_i^*)\eta_i^2 + \text{const.} \end{aligned} \tag{13}$$

which defines the coefficients c_i and d_i (both depending on η_i^*). Alternatives to the Taylor expansion are possible, like defining c_i and d_i as those minimizing the mean square error of the approximation in some interval ($\propto 1/\sqrt{\kappa}$) around η_i^* (Rue, 2001). This approach usually improves the approximation and is used for model 3. Hence, our GMRF proposal density for $\boldsymbol{\eta}$, is a GMRF with precision matrix $\mathbf{Q} = \kappa\mathbf{K} + \text{diag}(e_i d_i(\eta_i^*))$ and $\mathbf{b} = (\dots, y_i - e_i c_i(\eta_i^*), \dots)^T$, see (1). Let this density be denoted by $\tilde{\pi}(\boldsymbol{\eta}|\kappa, \mathbf{y}, \boldsymbol{\eta}^*)$.

Let $\boldsymbol{\eta}'$ be the current state and $\boldsymbol{\eta}''$ the new proposal. We choose $\boldsymbol{\eta}^* = \boldsymbol{\eta}'$. The proposal $\boldsymbol{\eta}''$ is then accepted with probability

$$\min \left\{ 1, \frac{\pi(\boldsymbol{\eta}''|\kappa, \mathbf{y})}{\pi(\boldsymbol{\eta}'|\kappa, \mathbf{y})} \frac{\tilde{\pi}(\boldsymbol{\eta}'|\kappa, \mathbf{y}, \boldsymbol{\eta}'')}{\tilde{\pi}(\boldsymbol{\eta}''|\kappa, \mathbf{y}, \boldsymbol{\eta}')} \right\}.$$

In scheme 3, a joint proposal for κ and $\boldsymbol{\eta}$ is constructed by first sampling a proposal κ'' from a distribution proportional to $(\kappa'' + \kappa')/(\kappa''\kappa')$ on $[\kappa'/f, \kappa'f]$, where κ' denotes the current value and $f > 1$ is a tuning constant. (The proposal κ'' can easily be generated by multiplying the current value κ' with a variable z with density proportional to $1 + 1/z$ on $[1/f, f]$.) Note that this is a Metropolis proposal since the proposal ratio $\pi(\kappa'|\kappa'')/\pi(\kappa''|\kappa')$ equals unity. Subsequently we sample the proposal $\boldsymbol{\eta}''$ as in scheme 2 (given the proposed value κ'') and accept/reject the proposal $(\kappa'', \boldsymbol{\eta}'')$ jointly with probability

$$\min \left\{ 1, \frac{\pi(\boldsymbol{\eta}'', \kappa''|\mathbf{y})}{\pi(\boldsymbol{\eta}', \kappa'|\mathbf{y})} \frac{\tilde{\pi}(\boldsymbol{\eta}'|\kappa'', \mathbf{y}, \boldsymbol{\eta}'')}{\tilde{\pi}(\boldsymbol{\eta}''|\kappa', \mathbf{y}, \boldsymbol{\eta}')} \right\}.$$

In contrast to scheme 2, the proposal ratio now also involves the computation of the normalizing constant of the GMRF, which depends on κ' and κ'' respectively.