# Fast sampling from a Gaussian Markov random field using Krylov subspace approaches

D. P. Simpson, I. W. Turner and A. N. Pettitt

December 5, 2007

**Abstract**

Many applications in spatial statistics, geostatistics and image analysis require efficient techniques for sampling from large Gaussian Markov random fields (GMRFs). A suite of methods, based on the Cholesky decomposition, for sampling from GMRFs, sampling conditioned on a set of linear constraints, and computing the likelihood were presented by Rue (2001). In this paper, we present an alternate set of methods based on Krylov subspace approaches. These methods have the advantage of requiring far less storage than the Cholesky decomposition and may be useful in problems where computing a Cholesky decomposition is infeasible.

**Keywords**

Gaussian Markov random field, Lanczos decomposition, matrix functions, saddle point system

## 1 Introduction

Gaussian Markov random fields (GMRFs) are important models in applied statistics. They can be utilised to model spatially structured uncertainty, seasonal variation, and other trends in the data; and are common model components in spatial statistics, image analysis and modelling of binary and categorical data (see (Rue and Held, 2005; Pettitt et al., 2002; Besag and Higdon, 1999; Cressie, 1991; Gilks et al., 1998) and references therein). Gaussian Markov random fields are used in a variety of ways in these applications. For example, an improper GMRF is used as a prior distribution to model structured spatial uncertainty in disease mapping (Held et al., 2005), while samples from a GMRF, in conjunction with simulated annealing, are used in an optimisation algorithm that aims to align gel tracks with a reference database (Glasbey, 2006). Another potential application uses samples from a GMRF to approximate stock market paths under the Black-Scholes model for pricing exotic options (L'Ecuyer, 2004). GMRFs are commonly used in Bayesian modelling and, hence, inference is usually made using Markov Chain Monte-Carlo (MCMC) and such methods usually require tens of thousands or millions of samples to be drawn from the GMRF in question (Rue and Held, 2005). It follows that efficient methods for generating multiple samples from large GMRFs are required in applied Bayesian modelling. In this paper, we present a suite of Krylov subspace methods for sampling from

a large GMRF. The methods presented here are intended to complement those presented by Rue (2001).

A GMRF is defined as follows. Consider a cloud of $n$ discrete points $\mathcal{V}$ in a region $\mathcal{D} \in \mathbb{R}^d$. At each point $s_i \in \mathcal{V}$, define a neighbourhood $\mathcal{N}_i$ and a Gaussian random variable $y_i$. Furthermore, let the random vector $y = [y_i]$ be distributed according to a multivariate Gaussian distribution with covariance matrix $\Sigma$. The joint probability density function for the GMRF $y$ is the multivariate normal density function

$$p(y) \propto \exp\left(-\frac{1}{2}y^T A y + b^T y\right), \tag{1}$$

where $A$ is the precision matrix, that is $A$ is the inverse of the covariance matrix, and the mean is defined implicitly by $A\mu = b$. The Markov property imposes the sparsity pattern $A_{ij} = 0$ if $j \notin \mathcal{N}_i$ on $A$ (Rue, 2001). For a multivariate normal distribution, $A$ is required to be symmetric positive semi-definite (Rue and Held, 2005) and, in this paper, the stronger condition of positive definiteness is assumed.

One method for sampling from a GMRF is to form

$$y = A^{-1}b + x, \tag{2}$$

where $x$ is a sample from the zero-mean GMRF $X \sim \mathcal{N}(0, A^{-1})$ (Rue and Held, 2005). While the first term in (2), $A^{-1}b$, can be approximated using a conjugate gradient method (Saad, 2003), the general method for approximating the second term, $x$, is given in the following algorithm.

---

**Input**: The size of the GMRF $n$, the precision matrix $A$.
**Output**: A sample, $x$, from the zero-mean GMRF parameterised by $A$.

---

Sample a vector, $z$, of independently and identically distributed (i.i.d.) standard normal variables, i.e. $z_i \sim \mathcal{N}(0, 1)$.
Decompose $A = RR^T$.
The sample is given as the solution to $R^T x = z$.

---

**Algorithm 1**: The general method for sampling from a zero mean GMRF.

Clearly, the sampling method depends on the choice of $R$. Choosing $R$ to be the Cholesky triangle of $A$ leads to a sampling method due to Rue (2001). When implemented using nested-disection techniques, this method requires $\mathcal{O}(n^{3/2})$ flops (Rue, 2001). In this paper, we will mainly be concerned with the alternate choice $R = A^{1/2}$, which leads to calculations of the form $x = A^{-1/2}z$ (Ilić et al., 2004). Whereas Ilić, Turner and Pettitt investigated the use of low degree polynomials to approximate $x$, in this paper we will investigate Krylov subspace methods for sampling from GMRFs.

Krylov subspaces form the basis for a large number of modern iterative methods for the solution of large sparse linear algebra problems. The $m$-dimensional Krylov subspace generated by $A \in \mathbb{R}^{n\times n}$ and $z \in \mathbb{R}^n$ is given by $\mathcal{K}_m(A, z) = \text{span}\left\{z, Az, A^2z, \ldots, A^{m-1}z\right\}$. While the basis given in the definition is useful for theoretical results, in practice an orthogonal basis for $\mathcal{K}_m(A, z)$ is preferred. When $A$ is symmetric, an orthogonal basis for $\mathcal{K}_m(A, z)$

is generated using the Lanczos decomposition

$$AV_m = V_m T_m + \beta_m v_{m+1} e_m^T, \tag{3}$$

where the columns of $V_m \in \mathbb{R}^{n\times m}$ form an orthonormal basis for $\mathcal{K}_m(A, z)$, $T_m \in \mathbb{R}^{m\times m}$ is a small tridiagonal matrix, $\beta_m$ is a constant, $v_{m+1}^T V_m = 0$, and $e_m = (0, 0, \ldots, 0, 1)^T \in \mathbb{R}^m$ is the $m^{\text{th}}$ vector in the standard basis for $\mathbb{R}^m$. For a detailed discussion of the implementation of the Lanczos decomposition, see (Stewart, 2001).

There are several differences between methods based on the Lanczos decomposition and those based on the Cholesky decomposition. Krylov methods require significantly less storage than methods based on the Cholesky decomposition. On the other hand, it is significantly faster to compute the second and subsequent samples using the Cholesky decomposition method, whereas the Krylov subspace approximation requires approximately the same computational effort for each sample as the Lanczos approximation depends explicitly on $z$. In order to achieve this speedup, the method of Rue assumes that, if $A$ is updated from one sample to the next, it changes in such a way that its Cholesky decomposition can be easily updated. This assumption is not necessary when using Krylov subspace approaches, which allows the consideration of a larger class of models and iterative procedures. It can be seen from this discussion that the Krylov subspace methods presented in this paper are designed to complement the Cholesky decomposition approach, allowing the consideration of models that may be computationally infeasible using that approach. In this paper we focus on the Lanczos approximation to the inverse square root, and extend the considerations to methods for restarting and accelerating the convergence of this approximation.

An important and related problem is that of sampling from a class of GMRFs with singular precision matrices, known as intrinsic GMRFs. These are used in statistical modeling to remove trend components in data (Rue and Held, 2005), for example let $\tilde{A}$ be a symmetric positive semi-definite matrix with nullity$(\tilde{A}) = k$, let $\{n_1, n_2, \ldots, n_k\}$ be an orthonormal basis (ONB) for $\mathcal{N}(\tilde{A})$ and let $N = [n_1, n_2, \ldots, n_k]$. Then, for any $y \in \mathcal{N}(\tilde{A})$, the improper density is invariant to the addition of vectors in the nullspace of $\tilde{A}$. A number of examples of intrinsic GMRFs can be found in Chapter three of (Rue and Held, 2005).

For computational purposes, it is convenient to view intrinsic GMRFs as a special case of a GMRF conditioned on linear constraints. We will denote a GMRF conditioned on linear constraints by $x|Bx = c$, where $x$ is a proper GMRF with symmetric positive definite precision matrix $A$, and $B \in \mathbb{R}^{k\times n}$ is the matrix of constraints (Rue, 2001). To see this, let $x$ be a zero-mean GMRF with non-singular precision matrix $\tilde{A} + \alpha NN^T$, where $\alpha > 0$, then the density of $y|N^T y = 0$ is a GMRF with mean 0 and singular precision matrix $\tilde{A}$. Without loss of generality, we will assume throughout this paper that the constraint matrix has full row rank. In most applications the number of constraints $k \ll n$. The connection between this problem and the general theory of saddle point problems (Benzi et al., 2005) was discussed in (Simpson et al., 2006) and three methods that were presented in that paper are reviewed in Section 3.

The final ingredient required for a full suite of routines for iterative methods

based on GMRFs is a method for approximating the log-likelihood

$$l(x) = -\frac{n}{2}\log(2\pi) + \frac{1}{2}\log(\det(A)) - \frac{1}{2}x^T Ax. \tag{4}$$

Clearly, the most expensive operation required for the evaluation of the log-likelihood is the approximation of the determinant of a large, sparse, symmetric positive definite matrix. A Monte-Carlo method for approximating $\log(\det(A))$ was presented in (Bai et al., 1996) and, for completeness, this will be briefly outlined in Section 4.

The outline of the paper is as follows. A Lanczos method, based on approximating the matrix-vector product $A^{-1/2}z$, for sampling from a zero-mean GMRF is considered in Section 2. The problems of restarting and preconditioning matrix function approximations are considered in Sections 2.1 and 2.2. Section 3 surveys three complementary methods for sampling from a GMRF conditioned on linear constraints. The Gaussian quadrature method for computing the approximate likelihood, originally presented in (Bai et al., 1996), is briefly reviewed in Section 4. Finally, a case study is presented in Section 5.

# 2 A Lanczos method for sampling from a zero mean GMRF

In order to approximate the product of a general matrix function and a vector, $f(A)z$, several authors (Saad, 1992; Hochbruck and Lubich, 1997; van den Eshof et al., 2002; Chiu et al., 2002; van der Vorst, 1987; Frommer and Simoncini, 2006) use the Lanczos approximation

$$f(A)z \approx \|z\| V_m f(T_m) e_1, \tag{5}$$

where $e_1 = (1, 0, 0, \ldots, 0)^T$ is the first canonical vector in the standard basis for $\mathbb{R}^m$. In this section we will investigate the case $f(t) = t^{-1/2}$. In this algorithm, the inverse square root of the tridiagonal matrix $T_m \in \mathbb{R}^{m\times m}$, needs to be accurately approximated. As $T_m$ is SPD, this is usually performed using an eigendecomposition, however, rational approximations or special methods constructed specifically for the inverse square root can also be used (Saad, 1992; Sidje, 1998; Davies and Higham, 2004; Hale et al., 2007).

Unlike Krylov subspace methods for the solution of linear systems or eigenvalue problems, the Lanczos approximation to matrix functions does not come equipped with a natural residual and, therefore, there is no natural measure of the accuracy of the approximation. An error bound based on Theorem 6 in (van den Eshof et al., 2002) is presented here and numerical experimentation has shown it to describe the error decay quite well.

**Theorem 1** *Let $A$ be a symmetric positive definite matrix with smallest and largest eigenvalues denoted $\lambda_{min}$ and $\lambda_{max}$ respectively. Then*

$$\left\| A^{-1/2}z - \|z\|_2 V_m T_m^{-1/2} e_1 \right\|_2 \leq \lambda_{min}^{-1/2} \|r_m\|_2 , \tag{6}$$

*where $r_m$ is the residual after using $m$ iterations of the conjugate gradient method to solve $Ay = z$.*

*Proof.* See (Ilić et al., 2007b). ■

This bound is important because it provides a way to gauge the accuracy of the approximation to $A^{-1/2}z$ from $\mathcal{K}_m(A, z)$ based on the norm of the residual for solving the linear system $Ax = z$. This quantity can be calculated for little additional cost during the Lanczos algorithm (Saad, 2003) and can, therefore, be used to determine the subspace size $m$ that guarantees a certain accuracy in the solution.

It should be noted that the preceding discussion has tacitly assumed that the basis $V_m$ is orthogonal. It is, however, well known that in floating point arithmetic, the basis generated by the Lanczos decomposition quickly loses orthogonality (Stewart, 2001). A number of complicated procedures for maintaining the orthogonality of the Lanczos basis have been presented in the literature, and these are surveyed in (Stewart, 2001). Numerical tests have, however, indicated that the loss of orthogonality does not pose serious problems when computing the inverse square root.

## 2.1 Restarting Lanczos approximations to the inverse square root

While the Lanczos approximation converges superlinearly to $x = A^{-1/2}z$, it requires the storage of the full orthogonal basis $V_m$. In practice, for very large GMRFs, the storage of the basis may not be feasible, and in these cases a number of alternative approaches have been suggested. Popolizio and Simoncini (2006) suggest a two-pass strategy, namely building $T_m$ while only storing three basis vectors and then recomputing the basis vectors. This approach, however, requires twice the work of the standard Lanczos approximation. An alternative strategy can be based on stopping the iteration after $m$ steps and 'restarting' the approximation with a new Krylov subspace. This approach is commonly used in Krylov subspace methods for solving non-symmetric linear systems (Saad, 2003). There are two approaches for restarting general matrix functions presented in the literature. The first, due to Eiermann and Ernst (2006) is based on polynomial interpolation and is designed for general non-normal matrices. The second approach, due to Ilić, Turner, and Simpson (2007b) is based on a decomposition of the residual of the associated linear system $Ay = z$. This method is designed for symmetric matrices and an analogue of Theorem 1 has been proven for a class of functions that include the inverse square root. Implementation details can be found in (Ilić et al., 2007b).

## 2.2 Accelerating convergence of the Lanczos approximation to the inverse square root

While restarting the Lanczos approximation allows the user to specify the amount of storage that the method will require, this comes at the price of convergence speed. This slowdown is demonstrated in Figure 1. The slow convergence occurs due to the loss of information accrued in the discarded subspaces. A common technique to overcome this slowdown when solving linear systems is to use some form of preconditioning (Saad, 2003). Preconditioning matrix function approximations is a more delicate affair than preconditioning linear

systems and, as such, only two methods have been proposed in the literature. These methods are outlined in the following subsections.

### 2.2.1 Adaptive preconditioning

The bound in Theorem 1 strongly suggests that the behaviour of the Lanczos approximation is almost entirely determined by the spectrum of $A$. In fact, it is well known that as $m$ increases, $\mathcal{K}_m(A, z)$ contains increasingly accurate approximations to the invariant subspaces of $A$ (Stewart, 2001; Ilić and Turner, 2004). Therefore, one method for improving the convergence of the restarted Lanczos approximation is to augment some of the converged eigenvectors of $A$ onto the new Krylov subspace. This idea is known as adaptive preconditioning (see (Burrage and Erhel, 1998; Chapman and Saad, 1997) for the linear case and (Ilić et al., 2007a) for the Lanczos approximation case).

The ideas of this preconditioner are as follows. Let the columns of the matrix $Q_j \in \mathbb{R}^{n\times j}$ span an eigenspace of $A$ and let $\Lambda_j = Q_j^T A Q_j$ be the associated generalised Rayleigh quotient. We form the preconditioner

$$M_j^{-1} = \gamma Q_j \Lambda_j^{-1} Q_j^T + I - Q_j Q_j^T,$$

where $\gamma = \frac{1}{2}(\theta_{min} + \theta_{max})$ and $\theta_{min} = \rho(\Lambda_j^{-1})$ and $\theta_{max} = \rho(\Lambda_j)$. Then $A_j = AM_j^{-1}$ has the same eigenvectors of $A$ and the eigenvalues corresponding to $Q_j$ have been shifted to $\gamma$. It should be noted that, in practice, the columns of $Q_j$ will only span an *approximately* invariant subspace of $A$ and, as such, the previous discussion is invalid. We have found, however, in numerical experiments that, providing the eigenvector estimates are reasonably good, that this effect is negligible. The error bound in Theorem 1 suggests that the adaptive scheme should focus on the smallest eigenvalues. Furthermore, as numerous samples are required for an MCMC method, the adaptive preconditioner can be extended as more eigenpairs converge. For full implementation details see (Ilić et al., 2007a).

### 2.2.2 Shift-and-invert preconditioning

An alternate method for accelerating the convergence of the Lanczos approximation is to implement a 'shift-and-invert' scheme. These methods, arising from Krylov methods for solving the eigenvalue problem, attempt to approximate $A^{-1/2}z$ on the space $\mathcal{K}_m\left((A-\xi I)^{-1}, z\right)$. This method was introduced independently by Moret and Novati (2004) and Hochbruck and van den Eshof (2006). The Krylov subspace $\mathcal{K}_m\left((A-\xi I)^{-1}, z\right)$ is built using a preconditioned conjugate gradient method at each step. Clearly, in order for this method to be successful, it is necessary for it to converge in far fewer operations that the standard Lanczos approximation. Popolizio and Simoncini (2006) found that the convergence of the shift-and-invert Lanczos method is strongly dependent on the choice of shift parameter $\xi$. Furthermore, numerical experiments reported in that paper suggest that this method works best when the spectral interval of $A$, i.e. the smallest interval containing the spectrum of $A$, is large. This was confirmed by our own experiments. For a detailed description of the implementation of the shift-and-invert Lanczos method, as well as a discussion on the best choice of the shift parameter, refer to (Popolizio and Simoncini, 2006; Hochbruck and van den Eshof, 2006).

# 3 Sampling from a GMRF conditioned on linear constraints

Given a sample from the unconditional GMRF $x$ from $\mathcal{N}(0, A^{-1})$, a sample from the corresponding GMRF conditioned on the linear constraints $Bx = c$ can be calculated using the update formula (Simpson et al., 2006)

$$\tilde{x} = x - \delta x, \tag{7}$$

where $\delta x$ is the first $n$ entries the the solution to the linear equations

$$\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} \delta x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ Bx - c \end{pmatrix}. \tag{8}$$

In the remainder of this section, we will denote the block matrix in (8) as $\mathcal{A}$. Systems of this form arise from the Karush-Kuhn-Tucker conditions in constrained optimisation problems and, as such, we will refer to this system as the K.K.T. system. They also occur in the finite element literature, where they are referred to as saddle point systems (Benzi et al., 2005). The update equation given in (Rue and Held, 2005) can be recovered from equation (8) by applying Schur-complement reduction. This yields the conditioning by Kriging formula

$$\delta x = A^{-1}B^T \left(BA^{-1}B^T\right)^{-1} (Bx - c). \tag{9}$$

Noting that the term $X = A^{-1}B^T$ occurs twice in the update equation (9), Rue (2001) suggested the use of the Cholesky decomposition that had already been computed during the unconditional sampling to solve the matrix equation

$$AX = B^T. \tag{10}$$

The update was then calculated using the formula $\delta x = X(BX)^{-1}z$. Methods of this form are known as *segregated methods* for solving the K.K.T. system (Benzi et al., 2005). A second class of methods, known as *coupled methods*, attempt to solve for $y$ and $\delta x$ jointly (see (Benzi et al., 2005) for a survey of methods for solving (8)). In this section, we will review the two segregated methods and one coupled method presented in (Simpson et al., 2006) for calculating the correction.

## 3.1 Segregated method 1: A multiple Krylov subspace approach

A direct extension of the method presented in (Rue, 2001) is to solve the $k$ linear systems $AX_{*i} = b_i$ using Krylov subspace methods, where $X_{*i}$ is the $i^{\text{th}}$ column of $X$ and $B^T = [b_1, \ldots, b_k]$. Simpson, Turner, and Pettitt (2006) showed that, if the desired solution accuracy is $\epsilon$, then the solution of each linear system needs to satisfy $\left\| r_m^{(i)} \right\| \leq \frac{\epsilon}{\sqrt{k}}$. The following algorithm summarises this method for calculating the correction.

**Input**: The size of the GMRF $n$, the precision matrix $A$, the constraint matrix $B^T$ and a tolerance $\epsilon$.
**Output**: The constraint correction $\delta x$ and $X = A^{-1}B^T$.

---

**for** $i = 1, 2, \ldots, k$ **do**
  Solve $AX_{*i} = b_i$ using the preconditioned conjugate gradient method until $\left\| r_m^{(i)} \right\| \leq \frac{\epsilon}{\sqrt{k}}$ (see Saad, 2003, for details).
  Form $X_i = \|b_i\|_2 V_m T_m^{-1} e_1$
**end**
Form $S = BX$ and solve $Sw = z$
Set $\delta x = Xw$.

**Algorithm 2**: A sequential Krylov subspace method for calculating the correction to a zero-mean GMRF conditioned on linear constraints.

## 3.2 Segregated method 2: A band Lanczos approach

In most applications, the number of constraints is significantly smaller than the size of the GMRF. Therefore, it is possible to use the *block* Krylov subspace $\mathcal{K}_m(A, B^T)$ to build an approximation to $X$. In general, this method requires fewer matrix-vector products (the dominant computational cost in Krylov methods) than the first method, however, it usually requires more storage. Implementation details can be found in (Simpson et al., 2006) and they are outlined in the following algorithm.

**Input**: The size of the GMRF $n$, the precision matrix $A$ and the constraint matrix $B^T$.
**Output**: The constraint correction $\delta x$ and $X = A^{-1}B^T$.

---

Set $R = B^T$.
**repeat**
  Compute QR-decomposition $R = QW$.
  Use Ruhe's variant of the Block Lanczos Method (Saad, 2003) to form $AU_m = U_m T_m + V_{m+1} T_{m+1,m} E_m^T$.
  Calculate $Y = T_m^{-1} \begin{pmatrix} W \\ 0 \end{pmatrix}$.
  Set $X = X + U_m Y$.
  Set $R = V_{m+1} T_{m+1,m} E_m^T Y$.
**until** *convergence criterion is met (see (Saad, 2003) for details)* ;
Form $S = BX$ and solve $Sw = z$.
Set $\delta x = Xw$.

**Algorithm 3**: A block-Lanczos method (Ruhe's variant) for calculating the correction to a zero-mean GMRF conditioned on linear constraints.

### 3.2.1 Method 3: A coupled approach

The final method that was outlined in Simpson et al. (2006) solves the full K.K.T. system $\mathcal{A}v = b$ using a Krylov subspace method. This is computationally feasible as the product of the K.K.T. matrix $\mathcal{A}$ with a vector can be computed in, at most, $C(A) + 2kn$ flops, where $C(A)$ is the cost of the matrix vector product involving $A$. Unfortunately, as $\mathcal{A}$ is not positive definite, the conjugate gradient method cannot be used on this system. In its place, we use an algorithm known as MINRES due to Paige and Saunders (1975), which finds the optimal approximation from $\mathcal{K}_m(\mathcal{A}, b)$. Further details on the coupled approach can be found in (Simpson et al., 2006; Benzi et al., 2005).

## 3.3 Recommendations

It was concluded in (Simpson et al., 2006) that all three methods can be useful in different situations. The fastest method for producing a single correction is method three, however, each subsequent correction requires the same amount of work as the first. On the other hand, as the segregated methods approximate the full matrix $X$, the time required to calculate subsequent samples is negligible compared to the time required to approximate $X$. When selecting a segregated method, one must decide whether less storage or fewer matrix-vector products are preferred as, in general, method one requires less storage than method two, whereas method two requires fewer matrix-vector products. If accuracy is not important, for example if the samples are then thresholded (as in (Pettitt et al., 2002)), or if $A$ changes non-linearly during the MCMC iterations, then it may be cheaper to use the coupled method - especially if a good preconditioner is available.

# 4 Approximate evaluation of the likelihood

In models where the matrix $A$ involves unknown parameters, in order to apply iterative techniques such as maximum likelihood estimation and MCMC procedures to estimate the posterior density of the GMRF, it is also necessary to approximate the determinant of a large sparse symmetric positive definite matrix (Rue, 2001). This problem has received a great deal of attention in the literature. Methods have been proposed that use various techniques including sparse approximate inverses (Reusken, 2001), Gauss quadrature (Bai et al., 1996) and diagonal approximations (Ipsen and Lee, 2003) to estimate $\det(A)$. It appears that the Gauss quadrature scheme developed by Bai, Fahey, and Golub (1996) is the most popular of these methods. This method is based on using Gauss quadrature to approximations the bilinear form $z^T \log(A) z$, and the use of a Monte Carlo method to approximate the identity

$$\log(\det(A)) = \mathrm{tr}(\log(A)). \tag{11}$$

This approach generates a confidence interval that contains $\det(A)$ with a set probability (Bai et al., 1996).

The crux of Bai, Fahey and Golub's method for estimating the determinant of an SPD matrix $A$ is the following result from (Hutchinson, 1990).

**Theorem 2** *((Hutchinson, 1990, Proposition 1)) Let $B \in \mathbb{R}^{n\times n}$ be a symmetric matrix with non-zero trace. Let $Z$ be the discrete random variable which takes the values $-1, 1$ each with probability $1/2$ and let $z = (z_1, z_2, \ldots, z_n)^T$ be a vector of $n$ independent samples from $Z$. Then $z^T Bz$ is an unbiased estimator of $\mathrm{tr}(B)$ and*

$$\mathrm{Var}(z^T Bz) = 2\sum_{i\neq j} b_{ij}^2.$$

*Moreover, $Z$ is the unique random variable amongst zero mean random variables for which $z^T Bz$ is a minimum variance, unbiased estimator of $tr(B)$.*

Therefore, (11) can be rewritten as

$$\log(\det(A)) = E(z^T \log(A)z).$$

Although this reformulation appears to have replaced an $n$ term sum with a $2^n$ term sum, it turns out that the expectation can be approximated using a Monte-Carlo rule with fewer than $n$ terms. Therefore, the calculation of the determinant has been reduced to approximating the bilinear form $z^T \log(A)z$.

The naive approach to approximating the bilinear form $z^T \log(A)z$ is to use the Lanzcos approximation to form

$$\begin{aligned} z^T \log(A)z &\approx z^T V_m \log(T_m) V_m^T z \\ &= n e_1^T \log(T_m) e_1, \end{aligned} \tag{12}$$

where $V_m$ is an ONB for $\mathcal{K}_m(A, z)$, and $z$ is a sample from $Z$. This approximation is relatively inexpensive as it does not require the storage of $V_m$ and can, therefore, be formed without considering restart and preconditioning procedures. This approximation was improved by Bai, Fahey and Golub, who exploited the connection between the Lanczos procedure and orthogonal polynomials to construct a Gauss quadrature scheme to find upper and lower bounds for $z^T \log(A)z$. This was achieved by modifying $T_m$ in (12) in an appropriate way, which allowed for the construction of confidence intervals for the true determinant as well as point estimates. Full details can be found in (Bai et al., 1996).

# 5 Case study — a simulation experiment

All tests were performed on a 2.33GHz Intel Core 2 Duo processor Macbook Pro using Matlab 7.4.0.287 (R2007a).

The covariance function that will be considered in the following case studies was introduced by Pettitt et al. (2002). Let $d_{ij} = d(s_i, s_j)$ be the distance between sites $s_i$ and $s_j$ in $\mathcal{V}$. The dependence between two nodes is defined by the function

$$\gamma_{ij} = \begin{cases} 1, & j \in \mathcal{N}_i^\delta \\ 0, & \text{otherwise,} \end{cases}$$

where $\mathcal{N}_i^\delta = \{j \in \mathbb{N} : s_j \in \mathcal{V}, d(s_i, s_j) < \delta, i \neq j\}$ and $\delta$ is the critical distance parameter which controls the sparsity of the precision matrix (Pettitt et al., 2002). The precision matrix $A$ can be related to the matrix $\gamma = [\gamma_{ij}]$ by the relation

$$A = I + \phi(D - \gamma), \tag{13}$$

where $I$ is the $n \times n$ identity matrix, $D = \text{diag}\left(\sum_{k \in \mathcal{N}_\delta^i} \gamma_{ik},\, i = 1, 2, \ldots, n\right)$, and $\phi > 0$ is a spatial dependence parameter (Pettitt et al., 2002). Elementwise, this is equivalent to

$$A_{ij} = \begin{cases} 1 + \phi \sum_{k \in \mathcal{N}_\delta^i} \gamma_{ik} & i = j, \\ -\phi \gamma_{ij} & i \neq j. \end{cases}$$

Clearly $A$ is symmetric and $A_{ii} > \sum_{j=1 j \neq i}^{n} |A_{ij}|$ for all $i = 1, \ldots, n$. It then follows by Theorem 12.2.16 in (Graybill, 1983) that $A$ is positive definite. It can also be shown that the smallest eigenvalue of $A$ is 1 (Ilić et al., 2004).

For this case study 1000 points were generated uniformly $[0, 5] \times [0, 5]$. These points were used to form a GMRF $y \sim \mathcal{N}(0, A^{-1})$, where the precision matrix $A$ is defined in (13). In this case study we will attempt to recover the parameters used to define $A$ by a MCMC process. In particular, we will focus on the neighbourhood size parameter $\delta$. We take for the prior distribution of $\delta$ a uniform distribution supported on $[0, 0.5]$ and the true parameter value is taken to be $\delta = 0.1$.

We performed the inference using a random walk Metropolis-Hasting algorithm and the proposal distribution was chosen to be uniform on

$$\left[\max\left(\delta - 10^{-3}, 0\right), \min\left(\delta + 10^{-3}, 0.5\right)\right].$$

At each step of this algorithm, it is necessary to calculate the ratio of the likelihoods, which is given by

$$2 \log\left(\frac{p(x|\delta^*)}{p(x|\delta)}\right) = \log(\det(A(\delta^*))) - \log(\det(A(\delta))) + x^T(A(\delta) - A(\delta^*))x.$$

The posterior for this model, generated using 5000 M-H iterations, is shown in Figure 2. The first 1000 samples were used as burn in.

As the matrix $A(\delta)$ is reasonably small, the inference could be carried out using Rue's method. The time required to calculate one determinant using Rue's method was $0.11s$, while the time required to calculate one determinant using the method of Bai, Fahey and Golub was $0.14s$. When comparing these two timings, however, it should be noted that the Cholesky decomposition requried for Rue's method was calculated using the `chol` command in Matlab. As this is a built in function, it should be expected that it runs much faster than a standard m file. To further this comparison, Table 1 gives a comparison of times for approximating the determinant and computing a sample for a GMRF with 8000 points and the same parameters. It can be seen that there is significant speed up gained by using the Krylov methods on this problem.

# Acknowledgements

The authors would like to thank Dr Miloš Ilić for his insightful comments on the methods presented in Section 2, Professor Gene Golub for pointing out the link between Schur compliment reduction and the K.K.T. system and Professor Håvard Rue for his comments, suggestions and support.

# References


Z. Bai, M. Fahey, and G. Golub. Some large scale matrix computation problems. *J. Computational and Applied Maths*, 74:71–89, 1996.

M. Benzi, G. Golub, and J. Liesen. Numerical solutions of saddle point problems. *Acta Numerica*, 14:1–137, 2005.

J. E. Besag and D. Higdon. Bayesian analysis of agricultural field experiments (with discussion). *J. Roy. Statist. Soc. B*, 61:691–746, 1999.

K. Burrage and J. Erhel. On the performance of various adaptive preconditioned GMRES strategies. *Numerical Linear Algebra with Applications*, 5(2):101–121, 1998.

A. Chapman and Y. Saad. Deflated and augmented Krylov subspace techniques. *Numerical Linear Algebra with Applications*, 4(1):43–66, 1997.

T.W. Chiu, T.H. Hsieh, C.H. Huang, and T.R. Huang. Note on the Zolotarev optimal rational approximation for the overlap Dirac operator. *Physical Review D*, 66:114502, 2002.

N.A. Cressie. *Statistics For Spacial Data*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, USA, 1991.

P. Davies and N. Higham. Computing $f(A)b$ for matrix function $f$. Technical Report 436, University of Manchester, 2004.

M. Eiermann and O. G. Ernst. A restarted Krylov subspace method for the evaluation of matrix functions. *SIAM J. Numer. Anal.*, 44(6):2481–2504, 2006.

A. Frommer and V. Simoncini. Matrix functions, 2006. URL `http://www.dm.unibo.it/ simoncin/fullpaper1.pdf`.

W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors. *Markov chain Monte Carlo in practice*. Chapman and Hall, 1998.

C.A. Glasbey. Warping of electrophoresis gels using generalisations of dynamic programming. In *Workshop on Interdisciplinary Statistics and Bioinformatics, Leeds*, 2006.

F. Graybill. *Matrices with Applications in Statistics*. Wadsworth and Brooks/Cole, California, USA, 2 edition, 1983.

N Hale, N.J. Higham, and L.N. Trefethen. Computing $A^\alpha$, $\log(A)$ and related matrix functions by contour integrals. *SIAM Journal of Numerical Analysis*, Submitted, 2007.

L. Held, I. Natario, S.E. Fenton, H. Rue, and N. Becker. Towards joint disease mapping. *Stat Methods Med Res.*, 14:61–82, 2005.

M. Hochbruck and C. Lubich. On Krylov subspace approximations to the matrix exponential operator. *SIAM J. Numer. Anal.*, 34(5):1911–1925, October 1997.

M. Hochbruck and J. van den Eshof. Preconditioning Lanczos approximations to the matrix exponential. *SIAM J. Sci. Comput.*, 27(4):1438–1457, 2006.

M. F. Hutchinson. A stochastic estimator of the trace ofthe influence matrix for laplacian smoothing splines. *Comm. Statist. Simula.*, 19(2):433–450, 1990.

M. Ilić and I. W. Turner. Krylov subspaces and the analytic grade. *Numer. Linear Algebra Appl.*, 12(1):55–76, 2004.

M. Ilić, I.W. Turner, and A.N. Pettitt. Bayesian computations and efficient algorithms for computing functions of large, sparse matrices. *ANZIAM J.*, 45 (E):C504–C518, 2004.

M. Ilić, I.W. Turner, and V. Anh. Numerical solution of the fractional Poisson equations using an adaptively preconditioned Lanczos method. *SIAM Journal on Numerical Analysis*, Submitted, 2007a.

M. Ilić, I.W. Turner, and D.P. Simpson. A restarted Lanczos approximation to functions of a symmetric matrix. *IMA Journal of Numerical Analysis*, Submitted, 2007b.

I. Ipsen and D. Lee. Determinant approximations. Technical Report CRSC-TR03-30, Centre for Research in Scientific Computing, 2003.

P. L'Ecuyer. Quasi-monte carlo methods in finance. In R. G. Incalls, M. D. Rosetti, J. S. Smith, and B. A. Peters, editors, *Proceedings of the 2004 Winter Simulation Conference*, 2004.

I. Moret and P. Novati. RD-rational approximation of the matrix exponential. *BIT Numerical Mathematics*, 44(3):595–615, 2004.

C. C. Paige and M. A. Saunders. Solution of sparse indefinite systems of linear equations. *SIAM J. Numerical Analysis*, 12:617–629, 1975.

A.N. Pettitt, I.S. Weir, and A.G. Hart. A conditional autoregressive Gaussian process for irregularly spaced multivariate data with application to modelling large sets of binary data. *Statistics and Computing*, 12(4):353–367, October 2002.

M. Popolizio and V. Simoncini. Acceleration techniques for approximating the matrix exponential operator. Technical report, Tech. Report, Dipartimento di Matematica, Universita di Bologna, 2006.

A. Reusken. Approximation of the determinant of large sparse symmetric positive definite matrices. *SIAM J. Matrix Anal. Appl.*, 23(3):799–818, 2001.

H. Rue. Fast sampling of Gaussian Markov random fields. *J. R. Statist. Soc. B*, 63:325–338, 2001.

H. Rue and L. Held. *Gaussian Markov Random Fields: Theory and Applications*. Chapman & Hall/CRC, 2005. USA.

Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, 2003.

Y. Saad. Analysis of some Krylov subspace approximations to the matrix exponential operator. *SIAM J. Numer. Anal.*, 29(1):209–228, February 1992.

R. B. Sidje. Expokit: A software package for computing matrix exponentials. *ACM Transactions on Math. Software*, 24:130–156, 1998.

D.P. Simpson, I.W. Turner, and A.N. Pettitt. Sampling from a Gaussian Markov random field conditioned on linear constraints. *ANZIAM J*, Submitted, 2006.

G. W. Stewart. *Matrix Algorithms, Volume 2: Eigensystems*. SIAM, 2001.

J. van den Eshof, A. Frommer, T. Lippert, K. Schilling, and H.A. van der Vorst. Numerical methods for the QCD overlap operator. I. sign-function and error bounds. *Computer Physics Communications*, 146:203–224, 2002.

H. A. van der Vorst. An iterative solution method for solving $f(A)x = b$ using Krylov subspace information obtained for the symmetric positive definite matrix $A$. *J. Computational and Applied Mathematics*, 18:249–263, 1987.

**Address**

D.P. Simpson*, I.W. Turner and
A.N. Pettitt,
School of Mathematical Sciences,
Queensland University of Technology,
Brisbane, Australia 4000.
Email: `dp.simpson@qut.edu.au`

* – Corresponding Author
A.N. Pettitt,
Department of Mathematics and Statistics,
Fylde College, Lancaster University,
Lancaster, LA1 4YF, United Kingdom

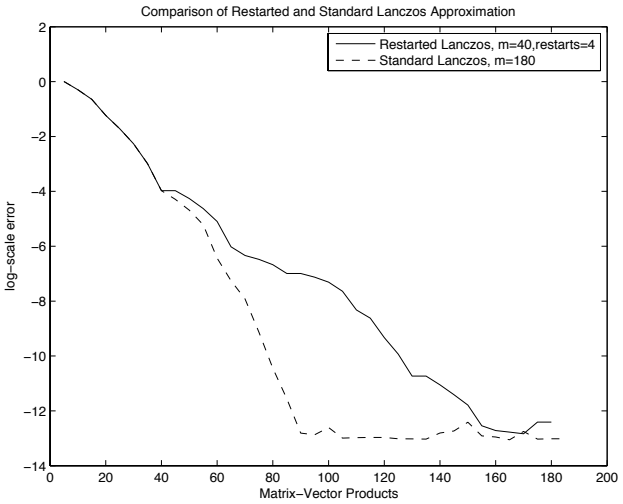


Figure 1: The convergence of the restarted Lanczos approximation (solid line) and the unrestarted Lanczos approximation (dashed line). This graph demonstrates both the superlinear convergence of the Lanczos method and the linear convergence of the restarted approximation.

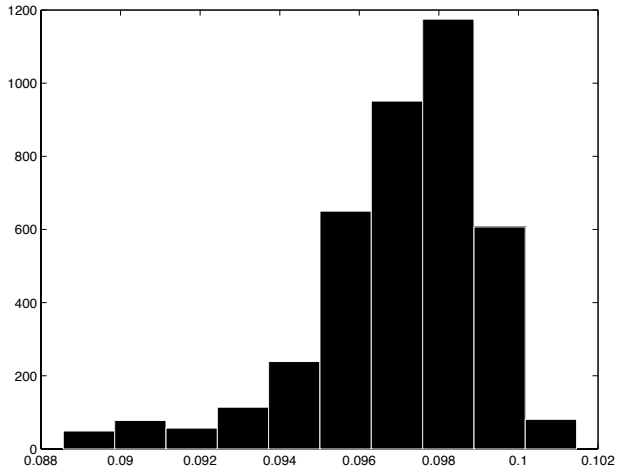


Figure 2: Histogram of the 4000 samples from the posterior $p(\delta|x)$ generated using random-walk Metropolis-Hastings.

Table 1: A comparison of execution times for $n = 8000$. When computing the sample, the subspace was increased until the target accuracy was achieved using Theorem 1. Three hundred iterations of the algorithm of Bai, Fahey, and Golub (1996) were used with $m = 30$.

| **Operation** | **Method** | **Time (seconds)** | Error |
|---|---|---|---|
| Sampling | Rue | 18.11 | — |
| | Krylov | 0.34 | $\leq 1.75e-9$ |
| Log-determinant | Rue | 17.49 | — |
| | Krylov | 8.98 | $6.28e-6$ (relative) |