NORGES TEKNISK-NATURVITENSKAPELIGE
UNIVERSITET

Specifying Gaussian Markov Random Fields with Incomplete Orthogonal Factorization using Givens Rotations

by

Xiangping Hu, Daniel Simpson and Håvard Rue

PREPRINT
STATISTICS NO. 2/2013

NORWEGIAN UNIVERSITY OF SCIENCE AND
TECHNOLOGY
TRONDHEIM, NORWAY

Specifying Gaussian Markov Random Fields with Incomplete Orthogonal Factorization using Givens Rotations

Xiangping Hu[*1], Daniel Simpson[1], and Håvard Rue[1]

[1]Department of Mathematical Sciences, Norwegian University of Science and Technology, N-7491 Trondheim, Norway

July 4, 2013

Abstract

In this paper an approach for finding a sparse incomplete Cholesky factor through an incomplete orthogonal factorization with Givens rotations is discussed and applied to Gaussian Markov random fields (GMRFs). The incomplete Cholesky factor obtained from the incomplete orthogonal factorization is usually sparser than the commonly used Cholesky factor obtained through the standard Cholesky factorization. On the computational side, this approach can provide a sparser Cholesky factor, which gives a computationally more efficient representation of GMRFs. On the theoretical side, this approach is stable and robust and always returns a sparse Cholesky factor. Since this approach applies both to square matrices and to rectangle matrices, it works well not only on precision matrices for GMRFs but also when the GMRFs are conditioned on a subset of the variables or on observed data. Some common structures for precision matrices are tested in order to illustrate the usefulness of the approach. One drawback to this approach is that the incomplete orthogonal factorization is usually slower than the standard Cholesky factorization implemented in standard libraries and currently it can be slower to build the sparse Cholesky factor.

Keywords: Gaussian Markov random field; Incomplete orthogonal factorization; Upper triangular matrix, Givens Rotation; Sparse matrix; Precision matrix

1 Introduction

Gaussian Markov random fields(GMRFs) are useful models in spatial statistics due to the Gaussian properties together with Markovian structures. They can also be formulated as conditional auto-regressions (CARs) models (Rue and Held, 2005). GMRFs have applications in many areas, such as spatial statistics, time-series models, analysis of longitudinal survival data, image analysis and geostatistics. See Rue and Held (2005,

---

[*]Corresponding author. Email: `Xiangping.Hu@math.ntnu.no`

Chapter 1) for more information and literature on how the GMRFs can be applied in different areas. From an analytical point of view GMRFs have good properties and can be specified through mean values $\boldsymbol{\mu}$ and covariance matrices $\boldsymbol{\Sigma}$. While from a computational point of view GMRFs can conveniently specified through precision matrices $\boldsymbol{Q}$ (the inverse of the covariance matrices $\boldsymbol{\Sigma}$), which are usually sparse matrices. The numerical algorithms for sparse matrices can be exploited for calculations with the sparse precision matrices and hence fast statistical inference is possible (Rue, 2001). The numerical algorithms for sparse matrices can be applied to achieve fast simulation of the fields and evaluation the densities (mostly, log-densities) of GMRFs and GMRFs with conditioning on subset of variables or linear constraints. See Rue and Held (2005, Chapter 2) for further details. These algorithms can also be used to calculate the marginal variances (Rue, 2005), and they can be extended to non-Gaussian cases (Rue et al., 2004).

Precision matrices $\boldsymbol{Q}$ are commonly used to specify GMRFs. This approach is natural due to the sparsity patterns of the precision matrices in Markovian models. In many situations the Cholesky factors are required and are crucial for simulation and inferences with GMRFs, and the Cholesky factors are normally obtained with Cholesky factorization routines in standard libraries. See Rue and Held (2005, Chapter 2) for different simulation algorithms for GMRFs using Cholesky factors. In order to get an even sparser Cholesky factor, with the purposes of saving computational resources, Wist and Rue (2006) showed that the Cholesky factor from an incomplete Cholesky factorization can be much sparser than the Cholesky factor from the regular Cholesky factorization. However, they provided theoretical and empirical evidence showing that the representation of sparser Cholesky factor was fragile when conditioning the GMRF on a subset of the variables or on observed data. It means that the sparsity patterns of the sparser Cholesky factors are destroyed when some constraints or observed data are introduced and the computational cost increases. Additionally, the sparser Cholesky factor from the incomplete Cholesky factorization is only valid for a specific precision matrix. Their approach is illustrated in Figure 1 with Routine 1.

In this paper a different approach is chosen to solve the problem presented by Wist and Rue (2006) . The main idea is given in the Figure 1 with Routine 2. In this approach one rectangular matrix $\boldsymbol{A}$ is formulated,

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{L}_1^{\mathrm{T}} \\ \boldsymbol{L}_2^{\mathrm{T}} \end{pmatrix}. \tag{1}$$

It consists of the Cholesky factor $\boldsymbol{L}_1^{\mathrm{T}}$ from the precision matrix $\boldsymbol{Q}_1$ of a given GMRF and the Cholesky factor $\boldsymbol{L}_2^{\mathrm{T}}$ of the matrix $\boldsymbol{Q}_2$. The matrix $\boldsymbol{Q}_2$ can be the additional effect when the GMRF is conditioned on observed data or on a subset of the variables. Both $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ are lower triangular matrices.

An incomplete orthogonal factorization is then used to factorize the matrix $\boldsymbol{A}$ in Equation (1) to find the sparse Cholesky factor for specifying the GMRF. It is shown that by using this approach an upper triangular matrix $\boldsymbol{R}$ which is sparser than the
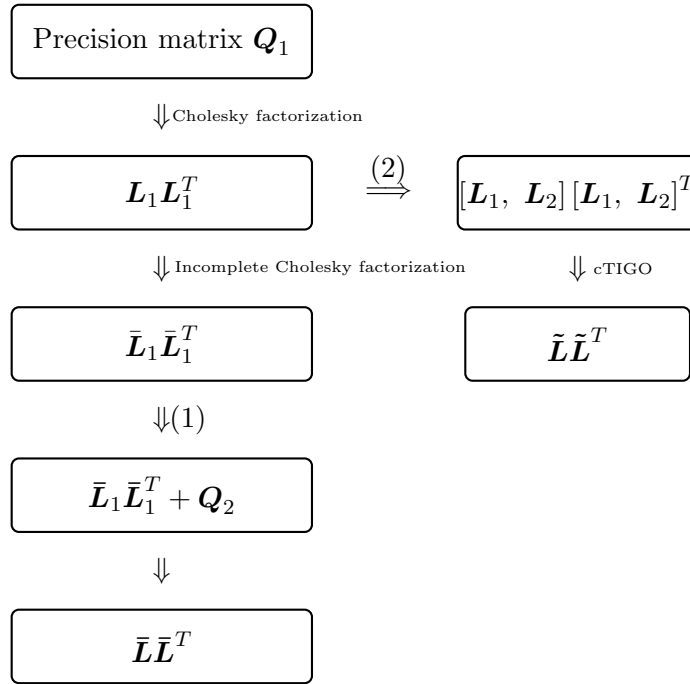
Figure 1: Diagram for the algorithm for finding sparser Cholesky factor by incomplete Cholesky factorization used by Wist and Rue (2006) (Routine 1) and the algorithm used in this paper (Routine 2).

standard Cholesky factor is obtained. Furthermore, this approach is applicable when the GMRF is conditioned on a subset of the variables or on observed data. Since the upper triangular matrix $\boldsymbol{R}$ is sparser in structure than the common Cholesky factor, it is better for applications.

The rest of this paper is organized as follows. In Section 2 some basic theory on GMRFs, sparsity patterns for precision matrices and Cholesky factors of GMRFs are presented. Some basic theories on the orthogonal factorization and the incomplete orthogonal factorization are also introduced in this section. In Section 3 the algorithm for obtaining the sparse Cholesky factor from the incomplete orthogonal factorization is introduced. A small example is given in order to illustrate how the algorithm works when the GMRFs are conditioned on a subset of the variables or on observed data. Results for different structures on the precision matrices are given in Section 4. Conclusion and general discussion in Section 5 ends the paper.

3

2 Background and Preliminaries

2.1 Basic theory on GMRFs

A random vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}} \in \mathbb{R}^n$ is called a GMRF if it is Gaussian distributed and processes a Markov property. The structure of a GMRF is usually presented by a labeled graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertexes $\{1, 2, \ldots, n\}$ and $\mathcal{E}$ is the set of edges. The graph $\mathcal{G}$ satisfies the properties that no edge between node i and node j if and only if $x_i \perp x_j | \boldsymbol{x}_{-ij}$ (Rue and Held, 2005), where $\{\boldsymbol{x}_{-ij}; i, j = 1, 2, \ldots, n\}$ denotes $\boldsymbol{x}_{-\{i,j\}}$. If the random vector $\boldsymbol{x}$ has a mean $\boldsymbol{\mu}$ and a precision matrix $\boldsymbol{Q}_1 > 0$, the probability density of the vector $\boldsymbol{x}$ is

$$\pi(\boldsymbol{x}|\boldsymbol{\theta}) = \left(\frac{1}{2\pi}\right)^{n/2} |\boldsymbol{Q}_1(\boldsymbol{\theta})|^{1/2} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{Q}_1(\boldsymbol{\theta})(\boldsymbol{x} - \boldsymbol{\mu})\right), \tag{2}$$

with the property

$$Q_{ij} \neq 0 \iff \{i, j\} \in \mathcal{E} \text{ for all } i \neq j.$$

The notation $\boldsymbol{Q}_1 > 0$ means that $\boldsymbol{Q}_1$ is a symmetric positive definite matrix. $\boldsymbol{\theta}$ denotes the parameters in the precision matrix. This implies that any vector with a Gaussian distribution and a symmetric positive definite covariance matrix is a Gaussian random field (GRF), and GMRFs are GRFs with Markov properties. The graph $\mathcal{G}$ determines the nonzero pattern of $\boldsymbol{Q}_1$. If $\mathcal{G}$ is fully connected, then $\boldsymbol{Q}_1$ is a complete dense matrix. A useful property of GMRF is that we can know whether x_i and x_j are conditionally independently or not directly from the precision matrix $\boldsymbol{Q}_1$ and the graph $\mathcal{G}$. Values of mean $\boldsymbol{\mu}$ do not have any influence on the pairwise conditional independence properties of the GMRFs, and hence we set $\boldsymbol{\mu} = \boldsymbol{0}$ in the following sections unless otherwise specified. The diagonal elements in the precision matrix $Q_{ii} \, (i = 1, 2, \ldots, n)$ are the conditional precisions of x_i given all the other nodes $\boldsymbol{x}_{-i}$. The off-diagonal elements $Q_{ij} \, (i, j = 1, 2, \ldots, n, i \neq j)$ can provide information about the correlations between x_i and x_j given on the nodes $\boldsymbol{x}_{-ij}$. These are the main differences in the interpretation between the precision matrix $\boldsymbol{Q}_1$ and the covariance matrix $\boldsymbol{\Sigma}_1$. The covariance matrix $\boldsymbol{\Sigma}_1$ contains the marginal variance of x_i and the marginal correlation between x_i and x_j. However, with the precision matrix the marginal properties are not directly available (Rue and Held, 2005).

Since $\boldsymbol{Q}_1$ is symmetric positive definite, there is a unique Cholesky factor $\boldsymbol{L}_1$ where $\boldsymbol{L}_1$ is a lower triangle matrix satisfying $\boldsymbol{Q}_1 = \boldsymbol{L}_1 \boldsymbol{L}_1^{\mathrm{T}}$. If we want to sample from the GMRF $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{Q}_1^{-1})$, the Cholesky factor $\boldsymbol{L}_1$ is commonly used. One algorithm for sampling GMRFs is given in Section 4.3. More algorithms for sampling GMRFs with different specifications are also available. See Rue and Held (2005, Chapter 2) for a detailed discussion on these algorithms. Rue and Held (2005) showed how to check the sparsity pattern of the Cholesky factor of a GMRF. Define

$$F(i, j) = i + 1, i + 2, \ldots, j - 1, j + 1, \ldots, n, \tag{3}$$

4

which is the future of i except j. Then

$$x_i \perp x_j \mid \boldsymbol{x}_{F(i,j)} \Longleftrightarrow L_{ji} = 0, \tag{4}$$

and $F(i,j)$ is called a separating subset of i and j. If $i \sim j$ denotes that i and j are neighbors, then $F(i,j)$ cannot be a separating subset for i and j whenever $i \sim j$. Further, the Cholesky factor of the precision matrix of a GMRF is always equally dense or denser than the lower triangle part of $\boldsymbol{Q}_1$.

In many situations there are more nonzero elements in $\boldsymbol{L}_1$ than in the lower triangular part of $\boldsymbol{Q}_1$. Denote $n_{\boldsymbol{L}_1}$ and $n_{\boldsymbol{Q}_1}$ the numbers of nonzero elements in the Cholesky factor $\boldsymbol{L}_1$ and the lower triangular part of precision matrix $\boldsymbol{Q}_1$, respectively. The difference $n_f = n_{\boldsymbol{L}_1} - n_{\boldsymbol{Q}_1}$ is called the fill-in. The ideal case is $n_f = 0$ or $n_{\boldsymbol{L}_1} = n_{\boldsymbol{Q}_1}$, but commonly $n_{\boldsymbol{L}_1} > n_{\boldsymbol{Q}_1}$ or even $n_{\boldsymbol{L}_1} \gg n_{\boldsymbol{Q}_1}$. It is known that the fill-in n_f not only depends on the graph, but also on the order of the nodes in the graph (Rue and Held, 2005). Thus a re-ordering is usually needed before doing a Cholesky factorization. It is desirable to find an optimal or approximately optimal ordering of the graph in order to make the Cholesky factor of $\boldsymbol{Q}_1$ sparser and to save computational resources, but this is not the focus in this paper. We refer to Rue and Held (2005) for more information on why it is desirable to do re-ordering of the graph of a GMRF.

2.2 Orthogonal factorization

With an $m \times n$ matrix $\boldsymbol{A}$, the orthogonal factorization of $\boldsymbol{A}$ is

$$\boldsymbol{A} = \boldsymbol{S} \cdot \boldsymbol{R}, \tag{5}$$

where $\boldsymbol{S} \in \mathbb{R}^{m \times m}$ is an orthogonal matrix and $\boldsymbol{R} \in \mathbb{R}^{m \times n}$ is an upper triangular matrix. We assume without loss of the generality that $m \geq n$. There exist many algorithms for orthogonal factorization, such as the standard Gram - Schmidt algorithm or the modified Gram–Schmidt (MGS) algorithm and the Householder orthogonal factorization. We refer to Saad (2003) and Björck (1996) for more algorithms. If $\boldsymbol{A}$ has full column rank, then the first n columns of $\boldsymbol{S}$ forms an orthonormal basis of ran($\boldsymbol{A}$), where ran($\boldsymbol{A}$) denotes the range of $\boldsymbol{A}$

$$\text{ran}(\boldsymbol{A}) = \{\boldsymbol{y} \in \mathbb{R}^m : \boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} \text{ for some } \boldsymbol{x} \in \mathbb{R}^n\}.$$

The orthogonal factorization is usually used to find an orthonormal basis for a matrix. The orthogonal factorization has many advantages and some of them are given in what follows.

1. It is numerically stable and robust both with a Householder orthogonal factorization and with a orthogonal factorization using Givens rotations. If the matrix $\boldsymbol{A}$ is non-singular, it always produces an orthogonal matrix $\boldsymbol{S}$ and an upper triangular matrix $\boldsymbol{R}$ which satisfy Equation (5);

2. It is easy to solve the linear system of equations $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ using the upper triangular matrix $\boldsymbol{R}$ since $\boldsymbol{S}$ is an orthogonal matrix;

3. The normal equation has the form $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{b}$ and the normal equation matrix is $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$, where $\boldsymbol{A}^{\mathrm{T}}$ denotes the transpose of $\boldsymbol{A}$. Then the triangular matrix $\boldsymbol{R}$ is the Cholesky factor of the normal equations matrix.

2.3 Givens rotations

A Givens rotation $G(i, j, \theta) \in \mathbb{R}^{m \times n}$ is an identity matrix $\boldsymbol{I}$ except that

$$
\begin{aligned}
G_{ii} &= c, & G_{ij} &= s, \\
G_{ji} &= -s, & G_{jj} &= c.
\end{aligned}
$$

If $c = \cos(\theta)$ and $s = \sin(\theta)$, then $\boldsymbol{y} = G(i, j, \theta) \cdot \boldsymbol{x}$ rotates $\boldsymbol{x}$ clockwise in the (i, j)-plane with θ radians, which gives

$$
y_l = \begin{cases}
x_l, & \text{when } l \neq i, j, \\
cx_i + sx_j, & \text{when } l = i, \\
-sx_i + cx_j, & \text{when } l = j.
\end{cases} \qquad (1 \leq l \leq m), \tag{6}
$$

If we want to rotate $\boldsymbol{x}$ counterclockwise in the (i, j)-plane with θ radians, then we can set $c = \cos(\theta)$ and $s = -\sin(\theta)$. It is obvious from Equation (6) that if

$$
s = \frac{x_j}{\sqrt{x_i^2 + x_j^2}} \text{ and } c = \frac{x_i}{\sqrt{x_i^2 + x_j^2}}
$$

then $y_j = 0$. So the Givens rotations can set the elements in $\boldsymbol{A}$ to zeros one at a time. This is useful when dealing with sparse matrices. At the same time, c and s are the only two values which we need for this algorithm. Givens rotations are suitable for structured least squares problems such as the problems at the heart of GMRFs.

2.4 Incomplete factorization algorithms

There are many algorithms for incomplete factorizations of matrices, such as the incomplete triangular factorization and the incomplete orthogonal factorization. These algorithms are commonly used in practical applications (Axelsson, 1996; Meijerink and van der Vorst, 1981; Saad, 1988). The incomplete factorizations usually have the form

$$
\boldsymbol{A} = \boldsymbol{M}_1 \cdot \boldsymbol{M}_2 + \boldsymbol{E}, \tag{7}
$$

where $\boldsymbol{E}$ is the error matrix, and $\boldsymbol{M}_1$ and $\boldsymbol{M}_2$ are some well-structured matrices. The incomplete factorization algorithms are usually associated with dropping strategies. A dropping strategy for an incomplete factorization specifies rules for when elements of the factors should be dropped. We returns to a detailed discussion on the dropping strategies in Section 3.5.

One of the commonly used incomplete factorization algorithms is the incomplete triangular factorization, and it is also called incomplete LU (ILU) factorization since

M_1 is a *lower* triangular matrix and M_2 is an *upper* triangular matrix. This algorithm is usually applied to the square matrices, and it uses Gaussian elimination together with a predefined dropping strategy. Many incomplete orthogonal factorizations can be used both for square matrices and for rectangular matrices, and these algorithms usually use the modified Gram-Schmidt procedure together with some dropping strategies in order to return a sparse and generally non-orthogonal matrix S and a sparse upper triangular matrix R. Wang et al. (1997) proved the existence and stability of the associated incomplete orthogonal factorization. Incomplete orthogonal factorization using Givens rotations was proposed by Bai et al. (2001). The main idea of the incomplete orthogonal factorization is to use the Givens rotations to zero-out the elements in the matrix one at a time. Some predefined dropping strategies are needed in order to achieve the sparsity pattern for the upper triangular matrix R. This algorithm computes a sparse matrix S, which is always an orthogonal matrix, together with a sparse upper triangular matrix R. Since the matrix S is the product of the Givens rotations matrices, it is always an orthogonal matrix. The incomplete orthogonal factorization has the form

$$A = S \cdot R + E. \tag{8}$$

This method was originally described and implemented by Jennings and Ajiz (1984). Saad (1988) described this incomplete orthogonal factorization with the modified Gram–Schmidt process using some numerical dropping strategy. Another version of the incomplete orthogonal factorization is given by Bai et al. (2001) with Givens rotations. Bai et al. (2001) claimed that this incomplete algorithm inherited the good properties of the orthogonal factorization.

1. R is a sparse triangular matrix and S is an orthogonal matrix. Bai et al. (2009) pointed out that the sparsity pattern of the upper-triangular part of A is inherited by the incomplete upper triangular matrix R. They also pointed out that the number of nonzero elements in the upper triangular matrix R is less than the number of nonzero elements in the upper-triangular part of A.

2. The error matrix $E = A - S \cdot R$ is "small" in some sense and the size of the errors can be controlled by the pre-defined threshold.

3. The triangular matrix R is non-singular whenever A is not singular. We can always obtain this triangular matrix in the same way as the orthogonal factorization and R will always be an incomplete Cholesky factor for the normal equation matrix $A^{\mathrm{T}}A$.

4. Another merit of the incomplete orthogonal factorization with Givens rotations is that we do not need to form the corresponding normal matrices S since only the (c, s)-pair is needed in order to find the upper triangular matrix R. More information about the Givens rotations and the (c, s)-pairs are given in Section 2.3

Papadopoulos et al. (2005) implemented different versions of the algorithm proposed by Bai et al. (2001). There are two main differences between these versions. The first

one is the order in which elements in the matrix $\boldsymbol{A}$ are zeroed out, and the second one is the rules for dropping strategies. We refer to Bai et al. (2001) and Papadopoulos et al. (2005) for more information about this algorithm and implementations. There are also more variations for incomplete orthogonal factorization using Givens rotations, such as Bai and Yin (2009) and Bai et al. (2009). Bai and Yin (2009) proposed some modified incomplete orthogonal factorization methods and these algorithms have special storage and sparsity-preserving techniques. Bai and Yin (2009) showed a way to adopt a diagonal compensation strategy by reusing the dropped elements. These dropped elements are added to the main diagonal elements of the same rows in the incomplete upper-triangular matrix $\boldsymbol{R}$. Bai et al. (2009) proposed practical incomplete Givens orthogonalization (IGO) methods for solving large sparse systems of linear equations. They claimed that these incomplete IGO methods took the storage requirements, the accuracy of the solutions and the coding of the pre-conditioners into consideration.

In this report, we have chosen the column-wise threshold incomplete Givens orthogonal (cTIGO) factorization algorithm for finding the sparse upper triangular matrix $\boldsymbol{R}$. This sparse upper triangular matrix $\boldsymbol{R}$ has sparse structure and can be used for specifying the GMRFs. The matrix $\boldsymbol{S}$ does not need to be stored in our setting since we only need the upper-triangular matrix $\boldsymbol{R}$. The matrix $\boldsymbol{S}$ only needs computed whenever it is explicitly needed.

3   Specifying GMRFs using sparse Cholesky factors

In this section we begin by introducing the background of GMRFs conditioned on a subset of the variables or on observed data. A small example is used to illustrate how the cTIGO algorithm works when applied to GMRFs.

3.1   GMRFs conditioned on a subset of the variables

I. GMRFs with soft constraint

Let $\boldsymbol{x}$ be a GMRF and assume that we have observed some linear transformation $\boldsymbol{Ax}$ with additional Gaussian distributed noise

$$\boldsymbol{e}|\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{Ax}, \boldsymbol{Q}_\epsilon^{-1}),$$

where k is the dimension of the vector $\boldsymbol{e}$, $\boldsymbol{A}$ is a $k \times n$ matrix with rank k and $k < n$, and $\boldsymbol{Q}_\epsilon > 0$ is the precision matrix of $\boldsymbol{e}$. This is called "soft constraint" by Rue and Held (2005) and the log-density for the model is

$$\log \pi(\boldsymbol{x}|\boldsymbol{e}) = -\frac{1}{2}(\boldsymbol{x}^{\mathrm{T}} - \boldsymbol{\mu})\boldsymbol{Q}_1(\boldsymbol{x} - \boldsymbol{\mu}) - \frac{1}{2}(\boldsymbol{e} - \boldsymbol{Ax})^{\mathrm{T}}\boldsymbol{Q}_\epsilon(\boldsymbol{e} - \boldsymbol{Ax}) + \mathrm{const}, \qquad (9)$$

where $\boldsymbol{\mu}$ and $\boldsymbol{Q}_1$ are the mean and the precision matrix of the GRMF, respectively, and "const" is constant. If $\boldsymbol{x}$ has mean $\boldsymbol{\mu} = \boldsymbol{0}$ then

$$\boldsymbol{x}|\boldsymbol{e} \sim \mathcal{N}_c(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{Q}_\epsilon\boldsymbol{e}, \boldsymbol{Q}_1 + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{Q}_\epsilon\boldsymbol{A}). \qquad (10)$$

Here we use the canonical form $\mathcal{N}_c(\cdot, \cdot)$ for $\boldsymbol{x}|\boldsymbol{e}$. We refer to Rue and Held (2005, Chapter 2.3.2) for more information about the canonical form for GMRF. We can notice that for specifying the GMRFs with "soft constraint", the Routine (2) as shown in Figure 1 can be applied since $\boldsymbol{Q} = \boldsymbol{Q}_1 + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{Q}_\epsilon\boldsymbol{A}$ with $\boldsymbol{Q_2} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{Q}_\epsilon\boldsymbol{A}$.

II. Models with auxiliary variables

Auxiliary variables are crucial in some models to retrieve GMRF full conditionals. We look at binary regression models with auxiliary variables.

Assume that we have Bernoulli observational model for binary responses. The binary responses have latent parameters which is a GMRF $\boldsymbol{x}$, and the GMRF usually depends on some hyperparameters $\boldsymbol{\theta}$. We usually choose the logit or probit models in this case, where

$$y_i \sim \mathcal{B}\left(\eta^{-1}(z_i^{\mathrm{T}}\boldsymbol{x})\right), \quad i = 1, 2, \ldots, m \tag{11}$$

where $\mathcal{B}(p)$ denotes a Bernoulli distribution with probability p for 1 and $1 - p$ for 0. $\boldsymbol{z}_i$ is a vector of covariates and we assume it is fixed. $\eta(\cdot)$ is a link function

$$\eta(p) = \begin{cases} \log\left(p/(1-p)\right) & \text{for logit link} \\ \Phi(p) & \text{for probit link} \end{cases} \tag{12}$$

where $\Phi(\cdot)$ denotes the cumulative distribution function (CDF) for standard Gaussian distribution. We can use models with auxiliary variables $\boldsymbol{\omega} = (\omega_1, \omega_2, \ldots, \omega_m)$ to represent these models,

$$\epsilon_i \stackrel{iid}{\sim} G(\epsilon_i),$$
$$\omega_i = \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{x} + \epsilon_i,$$
$$y_i = \begin{cases} 1, & \text{if } \omega_i > 0, \\ 0, & \text{otherwise,} \end{cases}$$

where $G(\cdot)$ is the CDF of standard logistic distribution in the logit case and $G(\cdot) = \Phi(\cdot)$ in the probit case. We refer to Forbes et al. (2011, Chapter 28) for more information about the standard logistic distribution and its CDF. Let $\boldsymbol{x}|\boldsymbol{\theta}$ be a GMRF of dimension n with mean $\boldsymbol{\mu} = \boldsymbol{0}$, and assume that we have $\boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{x} = x_i$ and $m = n$. With the probit link the posterior distribution is

$$\pi(\boldsymbol{x}, \boldsymbol{\omega}, \boldsymbol{\theta}|\boldsymbol{y}) \propto \pi(\boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\omega}|\boldsymbol{x})\pi(\boldsymbol{y}|\boldsymbol{\omega}). \tag{13}$$

The conditional distribution of $\boldsymbol{x}$ given the auxiliary variables can then be obtained

$$\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{\omega}) \propto \exp\left(-\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}_1(\boldsymbol{\theta})\boldsymbol{x} - \frac{1}{2}\sum_i(x_i - \omega_i)\right)^2,$$

and this can be written in the canonical form

$$\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{\omega} \sim \mathcal{N}_c(\omega, \boldsymbol{Q}_1(\boldsymbol{\theta}) + \boldsymbol{I}).$$

A general form for the conditional distribution of $\boldsymbol{x}$ given the auxiliary variables, for this binomial model with a probit link function, is given as

$$\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{\omega} \sim \mathcal{N}_c(\boldsymbol{Z}^{\mathrm{T}}\omega, \boldsymbol{Q}_1(\boldsymbol{\theta}) + \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Z}),$$

where $\boldsymbol{Z}$ is an $m \times n$ matrix. Similarly, the conditional distribution of $\boldsymbol{x}$ given the auxiliary variables for the logistic regression model can be written as

$$\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{\omega} \sim \mathcal{N}_c(\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{\Lambda}\omega, \boldsymbol{Q}_1(\boldsymbol{\theta}) + \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{\Lambda}\boldsymbol{Z}),$$

where $\boldsymbol{\Lambda} = \mathrm{diag}(\boldsymbol{\lambda})$, and λ_i is from the model specification. See more discussions on these models in Rue and Held (2005, Chapter 4.3).

In all the examples in this section, the models are suitable for use Routine (2) in Figure 1 to find the sparse Cholesky factors of the precision matrices of GMRFs.

3.2 GMRFs conditioned on data

As mentioned in Section 2.1, if a vector $\boldsymbol{x}$ is a GMRF with precision matrix $\boldsymbol{Q}_1$ and mean vector $\boldsymbol{\mu}$, then the density of the vector is given by Equation (2). In practical applications it is common to set $\boldsymbol{\mu} = \boldsymbol{0}$ (Rue and Held, 2005; Gneiting et al., 2010), which gives the probability density function

$$\pi(\boldsymbol{x}|\boldsymbol{\theta}) = \left(\frac{1}{2\pi}\right)^{n/2} |\boldsymbol{Q}_1(\boldsymbol{\theta})|^{1/2} \exp\left(-\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}_1(\boldsymbol{\theta})\boldsymbol{x}\right). \tag{14}$$

Assume that the data are of dimension k and defined as a k-dimensional random vector

$$\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta} \sim \mathcal{N}\left(\boldsymbol{A}\boldsymbol{x}, \boldsymbol{Q}_{\boldsymbol{\epsilon}}^{-1}\right)$$

and has the probability density function

$$\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta}) = \left(\frac{1}{2\pi}\right)^{k} |\boldsymbol{Q}_{\boldsymbol{\epsilon}}|^{1/2} \exp\left(-\frac{1}{2}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})^{\mathrm{T}}\boldsymbol{Q}_{\boldsymbol{\epsilon}}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})\right), \tag{15}$$

where $\boldsymbol{A}$ is a $k \times n$ matrix used to select the data location. The precision matrix $\boldsymbol{Q}_{\boldsymbol{\epsilon}}$ for the noise process is a positive definite matrix with dimension $k \times k$. Notice that the density function $\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$ is not dependent on the $\boldsymbol{\theta}$, and hence the probability density function $\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$ can be written as $\pi(\boldsymbol{y}|\boldsymbol{x})$. The probability density function of $\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}$ can be found from Equations (14) and (15) through

$$\begin{aligned} \pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) &\propto \pi(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{\theta}) \\ &= \pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta}) \\ &\propto \exp\left(-\frac{1}{2}\left[x^{\mathrm{T}}(\boldsymbol{Q}_1(\boldsymbol{\theta}) + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{Q}_{\boldsymbol{\epsilon}}\boldsymbol{A})\boldsymbol{x} - 2\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{Q}_{\boldsymbol{\epsilon}}\boldsymbol{y}\right]\right). \end{aligned} \tag{16}$$

Similarly, the density function (16) can be written in the canonical form as

$$\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta} \sim \mathcal{N}\left(\boldsymbol{\mu}_c(\boldsymbol{\theta}), \boldsymbol{Q}_c(\boldsymbol{\theta})\right). \tag{17}$$

where $\boldsymbol{\mu}_c(\boldsymbol{\theta}) = \boldsymbol{Q}_c(\boldsymbol{\theta})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{Q}_{\boldsymbol{\epsilon}}\boldsymbol{y}$, and $\boldsymbol{Q}_c(\boldsymbol{\theta}) = \boldsymbol{Q}_1(\boldsymbol{\theta}) + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{Q}_{\boldsymbol{\epsilon}}\boldsymbol{A}$. Now we can notice that the precision matrix for the GMRF conditional on data has the form $\boldsymbol{Q} = \boldsymbol{Q}_1(\boldsymbol{\theta}) + \boldsymbol{Q}_2$ with $\boldsymbol{Q}_2 = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{Q}_{\boldsymbol{\epsilon}}\boldsymbol{A}$, where $\boldsymbol{Q}_2$ does not depend on $\boldsymbol{\theta}$. Since $\boldsymbol{Q}$ has the same form as given in Routine (2) in Figure 1, it is possible to use the proposed routine to find the sparse Cholesky factor of the precision matrix of the GMRF conditioned on data.

Even though it is not the focus of this paper, it might be useful to point out that using Equations (14) - (16), we can find the analytical formula for the posterior density function of $(\boldsymbol{\theta}|\boldsymbol{y})$ through Bayes' formula. It is given by

$$\begin{aligned}
\log(\pi(\boldsymbol{\theta}|\boldsymbol{y})) = {}& \text{const.} + \log(\pi(\boldsymbol{\theta})) + \frac{1}{2}\log(|\boldsymbol{Q}_1(\boldsymbol{\theta})|) \\
& - \frac{1}{2}\log(|\boldsymbol{Q}_c(\boldsymbol{\theta})|) + \frac{1}{2}\boldsymbol{\mu}_c(\boldsymbol{\theta})^{\mathrm{T}}\boldsymbol{Q}_c(\boldsymbol{\theta})\boldsymbol{\mu}_c(\boldsymbol{\theta}).
\end{aligned} \tag{18}$$

We refer to Hu et al. (2012) for detailed information about this log-posterior density function. The log-posterior density function $\log(\boldsymbol{\theta}|\boldsymbol{y})$ is crucial when doing statistical inference in Bayesian statistics.

The sparse structure of $\boldsymbol{Q}_2$ depends both on the structures of $\boldsymbol{A}$ and of $\boldsymbol{Q}_{\boldsymbol{\epsilon}}$. In most cases, the $\boldsymbol{Q}_{\boldsymbol{\epsilon}}$ is a diagonal matrix and the matrix $\boldsymbol{A}$ has sparse structure. Therefore $\boldsymbol{Q}_2$ should also have a sparse structure. When the observations are conditional independent, but have a non-Gaussian distribution, then we can use a GMRF approximation to obtain a sparse structure of $\boldsymbol{Q}_2$ as presented in Section 3.3.

3.3 GMRF approximation

Suppose there are n conditionally independent observations $y_1, y_2, \ldots, y_n$ from a non-Gaussian distribution and that y_i is an indirect observation of x_i. $\boldsymbol{x}$ is a GMRF with mean $\boldsymbol{\mu} = \boldsymbol{0}$ and precision matrix $\boldsymbol{Q}_1$. The full conditional $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ then has the form

$$\pi\left(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}\right) \propto \exp\left(-\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}_1\boldsymbol{x} + \sum_{i=1}^{n}\log\pi(y_i|x_i)\right). \tag{19}$$

Apply a second-order Taylor expansion of $\sum_{i=1}^{n}\log\pi(y_i|x_i)$ around $\boldsymbol{\mu}_0$. In other words, construct a suitable GMRF proposal density $\widetilde{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$

$$\begin{aligned}
\widetilde{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) &\propto \exp\left(-\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}_1(\boldsymbol{\theta})\boldsymbol{x} + \sum_{i=1}(a_i + b_i x_i - \frac{1}{2}c_i x_i^2)\right) \\
&\propto \exp\left(-\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\left(\boldsymbol{Q}_1(\boldsymbol{\theta}) + \mathrm{diag}(\boldsymbol{c})\right)\boldsymbol{x} + \boldsymbol{b}^{\mathrm{T}}\boldsymbol{x}\right).
\end{aligned} \tag{20}$$

c_i should set to zero when $c_i < 0$. $\boldsymbol{b}$ and $\boldsymbol{c}$ depend on $\boldsymbol{\mu}_0$. The canonical parametrization of $\widetilde{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ has the form

11

$$\mathcal{N}_c \left(\boldsymbol{b}, \boldsymbol{Q}_1(\boldsymbol{\theta}) + \operatorname{diag}(\boldsymbol{c}) \right).$$

In this case $\boldsymbol{Q}_2$ has a diagonal structure. An important feature of (20) is that it inherits the *Markov* property of the prior on $\boldsymbol{x}$, which is useful for sampling GMRF. When $\boldsymbol{\mu} \neq \boldsymbol{0}$, the canonical parametrization of the $(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ is changed to

$$\mathcal{N}_c \left(\boldsymbol{Q}\boldsymbol{\mu} + \boldsymbol{b}, \boldsymbol{Q}_1(\boldsymbol{\theta}) + \operatorname{diag}(\boldsymbol{c}) \right),$$

and does not change the matrix $\boldsymbol{Q} = \boldsymbol{Q}_1 + \boldsymbol{Q}_2$.

As it was pointed out in Section 2.1, to sample from the GMRFs, the Cholesky factor $\boldsymbol{L}$ is one of most important factors. In order to save computational resources, a sparse Cholesky factor is preferable if the approximated precision matrix is "close" to the original precision matrix, where "close" means both in structure and the elements.

3.4   Theoretical background

It has been mentioned in Section 2.1 that the sparsity pattern of the Cholesky factor is determined by the graph $\mathcal{G}$, and it is unnecessary to calculate the zero elements in the Cholesky factor. In this section, we are going to introduce the theoretical background for finding the Cholesky factor from the orthogonal factorization when the GMRF is conditioned on observed data or a subset of the variables.

Let $\boldsymbol{y}$ be the observed data and assume $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$ has the Gaussian distribution, then the density of $\boldsymbol{x}$ conditioned on $\boldsymbol{y}$ has the form in (16). In the discussed situations in Section 3.1 - Section 3.3, the precision matrix $\boldsymbol{Q}$ can be split into two parts, the precision matrix $\boldsymbol{Q}_1$ of the GMRF $(\boldsymbol{x}|\boldsymbol{\theta})$ and the matrix $\boldsymbol{Q}_2$ which is the additional effect. The matrix $\boldsymbol{Q}_2$ is usually a diagonal matrix or another type of sparse matrix. If the data is not Gaussian distributed, then we can apply the GMRFs approximation given in (20) and it returns the precision matrix $\boldsymbol{Q}_1$ with a diagonal matrix $\boldsymbol{Q}_2$ added. This structure satisfies the Routine (2) in Figure 1.

Let $\boldsymbol{Q} = \boldsymbol{Q}_1 + \boldsymbol{Q}_2$ and assume that the Cholesky factors for $\boldsymbol{Q}_1$, $\boldsymbol{Q}_2$ and $\boldsymbol{Q}$ are $\boldsymbol{L}_1$, $\boldsymbol{L}_2$ and $\boldsymbol{L}$, respectively. The Cholesky factors $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ are assumed to be known. We have the following results.

Observation 1. *Let $\boldsymbol{x} \in \mathbb{R}^n$ be a zero mean GMRF with precision matrix $\boldsymbol{Q}_1$. Assume that the precision matrix has the form $\boldsymbol{Q} = \boldsymbol{Q}_1 + \boldsymbol{Q}_2 \in \mathbb{R}^{n \times n}$ when conditioned on observed data or a subset of the variables. Let the Cholesky factors for $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ be $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$, respectively. Form*

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{L}_1^T \\ \boldsymbol{L}_2^T \end{pmatrix}.$$

Then $\boldsymbol{A}^T \boldsymbol{A}$ is the precision matrix $\boldsymbol{Q}$.

Proof. $\boldsymbol{A}^{\mathrm{T}} \boldsymbol{A} = \begin{pmatrix} \boldsymbol{L}_1 & \boldsymbol{L}_2 \end{pmatrix} \begin{pmatrix} \boldsymbol{L}_1^{\mathrm{T}} \\ \boldsymbol{L}_2^{\mathrm{T}} \end{pmatrix} = \boldsymbol{L}_1 \boldsymbol{L}_1^{\mathrm{T}} + \boldsymbol{L}_2 \boldsymbol{L}_2^{\mathrm{T}} = \boldsymbol{Q}_1 + \boldsymbol{Q}_2 = \boldsymbol{Q}.$ $\qquad\qquad\square$

From Observation 1 the following corollaries are established. Sketched proofs for these corollaries are given. We refer to Simpson (2008) for numerical examples with Corollary 1.

Corollary 1. *Let $\boldsymbol{X}$ be a zero mean GMRF with precision matrix $\boldsymbol{Q} = \boldsymbol{Q}_1 + \boldsymbol{Q}_2 \in \mathbb{R}^{n \times n}$, and let $\boldsymbol{A}$ have the form given in Observation 1. Let $\boldsymbol{z} \in \mathbb{R}^{2n}$ be a vector of independent and identically distributed (i.i.d.) standard Gaussian random variables. Then the solution of the least squares problem*

$$\boldsymbol{x} = \arg\min_y \|\boldsymbol{A}\boldsymbol{y} - \boldsymbol{z}\|_2 \tag{21}$$

is a sample from the GMRF $\boldsymbol{X}$.

Proof. $\boldsymbol{Q} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ from Observation 1 is the starting point to prove this Corollary. Denote $\boldsymbol{A}^\dagger$ the Moore-Penrose pseudo-inverse of $\boldsymbol{A}$, and then the solution to the least squares problem is $\boldsymbol{x} = \boldsymbol{A}^\dagger \boldsymbol{z}$ (Björck, 1996). From the definition of the pseudo-inverse, $\boldsymbol{x} = \boldsymbol{W}\boldsymbol{S}^\dagger \boldsymbol{U}^{\mathrm{T}}\boldsymbol{z}$, where $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{W}^{\mathrm{T}}$ is a singular value decomposition of $\boldsymbol{A}$ and $\boldsymbol{S}^\dagger \in \mathbb{R}^{d \times 2d}$ is the matrix with the reciprocals of the non-zero singular values on the diagonal. We can verify that $\boldsymbol{x}$ has the required distribution, and it is sufficient to check the first two moments since $\boldsymbol{x}$ has a Gaussian distribution, being linear in $\boldsymbol{z}$. It is clear that $\mathbb{E}(\boldsymbol{x}) = 0$. Furthermore,

$$\begin{aligned}
\mathbb{E}(\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}) &= \boldsymbol{A}^\dagger \mathbb{E}(\boldsymbol{z}\boldsymbol{z}^{\mathrm{T}})(\boldsymbol{A}^\dagger)^{\mathrm{T}} \\
&= \boldsymbol{W}\boldsymbol{S}^\dagger \boldsymbol{U}^{\mathrm{T}}\boldsymbol{U}(\boldsymbol{S}^\dagger)^{\mathrm{T}}\boldsymbol{W}^{\mathrm{T}} \\
&= \boldsymbol{W}\boldsymbol{S}^\dagger (\boldsymbol{S}^\dagger)^{\mathrm{T}}\boldsymbol{W}^{\mathrm{T}} \\
&= \boldsymbol{W}(\boldsymbol{S}^{\mathrm{T}}\boldsymbol{S})^\dagger \boldsymbol{W}^{\mathrm{T}}.
\end{aligned}$$

Calculations yield $\boldsymbol{Q} = \boldsymbol{W}\boldsymbol{S}^{\mathrm{T}}\boldsymbol{S}\boldsymbol{W}^{\mathrm{T}}$ and, hence, $\boldsymbol{x} \sim MVN(0, \boldsymbol{Q}^\dagger)$. □

Therefore, it is possible to sample from a GMRF by solving the sparse least squares problem given in (21) with some conditions on GMRFs.

Corollary 2. *The upper triangular matrix $\boldsymbol{R}$ from the orthogonal factorization of the rectangular matrix*

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{L}_1^T \\ \boldsymbol{L}_2^T \end{pmatrix} \tag{22}$$

is the Cholesky factor of the precision matrix $\boldsymbol{Q} = \boldsymbol{Q}_1 + \boldsymbol{Q}_2$.

Proof. Since the upper triangular matrix from the orthogonal factorization is the Cholesky factor for the normal equations matrix, this is obvious from Observation 1. □

By using the orthogonal factorization of the rectangular matrix $\boldsymbol{A}$, it is possible to get samples from the GMRFs when they are conditioned on data or a subset of the variables by using Corollary 1 or the Cholesky factor from Observation 1 together with the sampling algorithms discussed in Rue and Held (2005, Chapter 2).

3.5 Dropping strategies

In this section the dropping strategy for the incomplete orthogonal factorization is introduced in order to find the incomplete Cholesky factor for matrix $A^{\mathrm{T}}A$. Together with some dropping strategy for the incomplete orthogonal factorization of the rectangular matrix A, a sparse upper triangular matrix R can be obtained. From Corollary 2 and the discussion in Section 2.2, we know that R is an incomplete Cholesky factor or sparse Cholesky factor for the precision matrix Q. This sparse Cholesky factor can then be used to specify the GMRF. The dropping strategies are important when doing the incomplete orthogonal factorization. Generally speaking, there are two kinds of dropping strategies.

1. Drop fill-ins based on sparsity patterns. Before doing the incomplete orthogonal factorization, the sparsity pattern of the upper triangular matrix is predefined and fixed. If the factorization based only on the sparsity pattern of the original matrix, we drop all the elements which are pre-defined to be zeros. The algorithm does not consider the actual numerical values of the elements during the factorizations.

2. Drop fill-ins by using a numerical threshold. This strategy only includes the elements in R if they are bigger than a predefined threshold value. Munksgaard (1980) presented one way to select the value of the threshold parameter. His strategy drops the elements which are smaller than the diagonal elements of their rows and columns, multiplied by some predefined small value (called dropping tolerance). In this report, a slightly different dropping strategy is chosen. During the incomplete orthogonal factorization using Givens rotations, or the column-wise threshold incomplete Givens orthogonal (cTIGO) factorization (Papadopoulos et al., 2005), we drop the elements according to their magnitudes with some predefined dropping tolerance. The nonzero pattern of R is determined dynamically.

Both the fixed sparsity pattern strategy and the dynamic strategy are useful in applications. The fixed sparsity pattern strategy is the candidate when the computation resources are low. It is usually faster but sometimes returns unsatisfactory results. The dynamic strategy will in most cases return satisfactory results by choosing proper dropping tolerances but it is usually more expensive both in time and computations.

There are different versions of orthogonal factorizations. We refer to Saad (2003), Golub and Van Loan (1996) and Trefethen and Bau (1997) for more information. Based on the research of Bai et al. (2001), Papadopoulos et al. (2005) and Bai et al. (2009), we choose the incomplete orthogonal factorization using Givens rotations to find the sparse Cholesky factor. This algorithm is stable and robust and always returns a sparse matrix. This algorithm inherits the advantages of orthogonal factorization. Bai et al. (2001) commented that there is little attention given to incomplete orthogonal factorization with Givens rotations, which is actually useful in many numerical problems.

In order to use Givens rotations for incomplete orthogonal factorization, the following nonzero patterns needs to be defined,

$$N_{\boldsymbol{Q}} = \{(i,j) \mid Q_{ij} \neq 0, 1 \leq i,j \leq n\},$$
$$N_{\boldsymbol{Q},l} = \{(i,j) \mid Q_{ij} \neq 0, i \geq j, 1 \leq i,j \leq n\},$$
$$N_{\boldsymbol{Q},u} = \{(i,j) \mid Q_{ij} \neq 0, i \leq j, 1 \leq i,j \leq n\},$$
$$N_{\boldsymbol{L}_1} = \{(i,j) \mid L_{1_{ij}} \neq 0, 1 \leq i,j \leq n\},$$
$$N_{\boldsymbol{L}_2} = \{(i,j) \mid L_{2_{ij}} \neq 0, 1 \leq i,j \leq n\},$$
$$N_{\boldsymbol{L}} = \{(i,j) \mid L_{ij} \neq 0, 1 \leq i,j \leq n\},$$
$$N_{\boldsymbol{A}} = \{(i,j) \mid A_{ij} \neq 0, 1 \leq i,j \leq n\},$$
$$N_{\boldsymbol{R}} = \{(i,j) \mid R_{ij} \neq 0, 1 \leq i,j \leq n\},$$

where $N_{\boldsymbol{Q}}$ is the nonzero pattern of the matrix $\boldsymbol{Q}$, and $N_{\boldsymbol{Q},u}$, $N_{\boldsymbol{Q},l}$ are the nonzero patterns of the upper and lower triangular parts of the matrix $\boldsymbol{Q}$, respectively. $N_{\boldsymbol{L}_1}$, $N_{\boldsymbol{L}_2}$, $N_{\boldsymbol{L}}$ $N_{\boldsymbol{A}}$ and $N_{\boldsymbol{R}}$ are the nonzero patterns of the lower triangular matrix $\boldsymbol{L}_1$, the lower triangular matrix $\boldsymbol{L}_2$, the lower triangular matrix $\boldsymbol{L}$, the matrix $\boldsymbol{A}$ and the matrix $\boldsymbol{R}$, respectively. These matrices are already formulated in previous sections.

In order to use the cTIGO algorithm, the rectangular matrix $\boldsymbol{A}$ in (22) is formed. The sparsity pattern of matrix $\boldsymbol{A}$ is already known beforehand. However, since the dynamic strategy is chosen, there will be some fill-in during Givens rotations process, and the sparsity pattern of the sparse Cholesky factor $\boldsymbol{R}$ will depend on the dropping tolerance and usually $N_{\boldsymbol{R}} < N_{\boldsymbol{A}}$. For more information about cTIGO algorithm, see Bai et al. (2001) for theoretical issues and Papadopoulos et al. (2005) for implementations.

3.6 A small example

A small example is explored in this section to illustrate how to use the cTIGO algorithm to find the sparse Cholesky factor $\boldsymbol{R}$. For simplicity and without loss of generality, we assume that $\boldsymbol{Q}_1$ is the precision matrix for a zero mean GMRF $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_1)$, and that the data are normally distributed, i.e., $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ and hence $\boldsymbol{Q}_2 = \boldsymbol{I}$. Assume that these matrices are given as follows

$$\boldsymbol{Q_1} = \begin{pmatrix} 5 & -1 & 0 & \dots & 0 & -1 \\ -1 & 5 & -1 & 0 & \dots & 0 \\ 0 & -1 & 5 & -1 & \dots & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ -1 & 0 & \dots & & -1 & 5 \end{pmatrix}_{9 \times 9}$$

and

$$\boldsymbol{Q_2} = \boldsymbol{I}_{9 \times 9} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \\ 0 & \dots & & & 1 \end{pmatrix}_{9 \times 9}.$$

15

Let $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ denote the Cholesky factor of the two matrices $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$, respectively, with the sparsity patterns given in Figure 2(a) and Figure 2(b). The rectangular matrix $\boldsymbol{A}$ can then be formed as given in (22) with the sparsity pattern given in Figure 2(d). Apply the cTIGO algorithm to the rectangular matrix $\boldsymbol{A}$ with a dropping tolerance of 0.0001 to find the sparse incomplete Cholesky factor $\boldsymbol{R}$. The sparsity pattern of $\boldsymbol{R}$ is given in Figure 2(e). The sparsity pattern of the Cholesky factor $\boldsymbol{L}$ from the standard Cholesky factorization of the precision matrix $\boldsymbol{Q}$ is given in Figure 2(c).
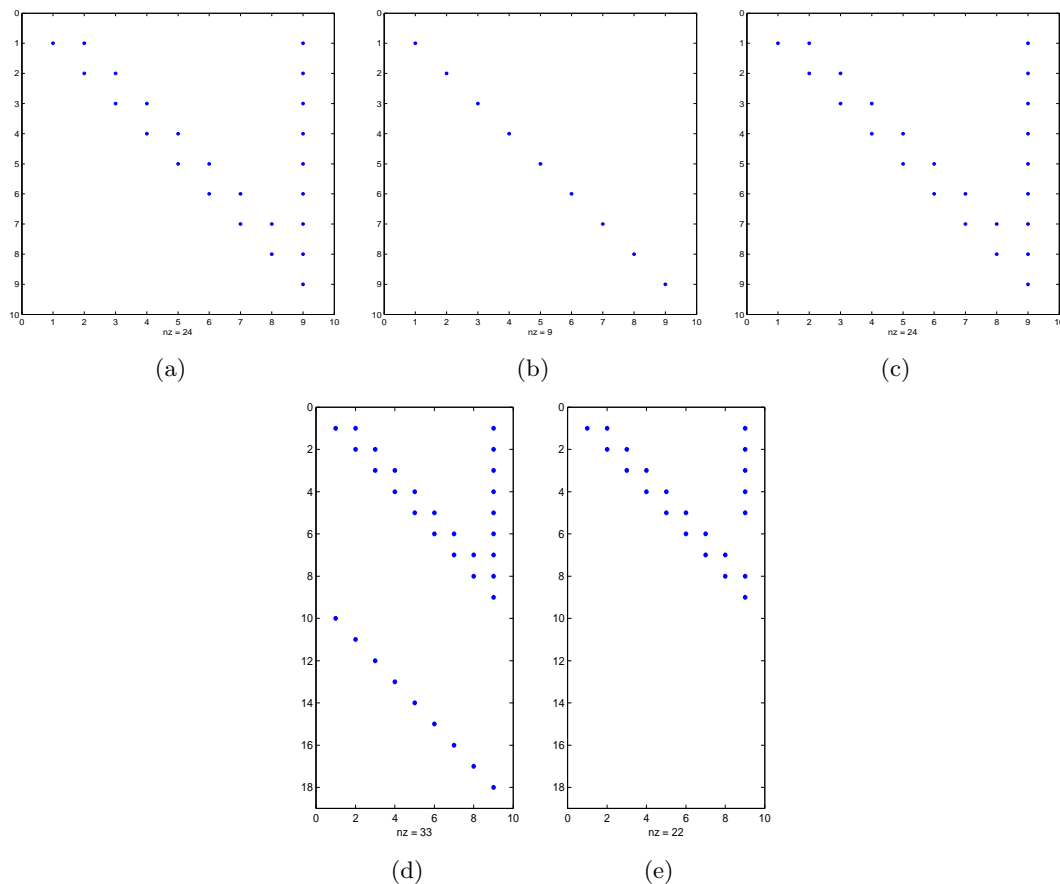
Figure 2: The sparsity patterns of $\boldsymbol{L}_1$ (a), $\boldsymbol{L}_2$ (b), $\boldsymbol{L}$ (c), $\boldsymbol{A}$ (d) and $\boldsymbol{R}$ (e)

We notice that the precision matrix $\boldsymbol{Q}_1$ is quite similar to the tridiagonal matrix except the values at two of the corners. However, there is a lot of fill-in in the Cholesky factor $\boldsymbol{L}_1$. This is a common structure for the precision matrix of a GMRF, for instance, a GMRF on a torus. The same comments can be given for $\boldsymbol{Q}$ and $\boldsymbol{L}$. Note that the upper triangular matrix $\boldsymbol{R}$ has less nonzero elements than $\boldsymbol{L}_1$, $\boldsymbol{L}$ and $\boldsymbol{A}$, $N_{\boldsymbol{R}} < N_{\boldsymbol{L}_1}, N_{\boldsymbol{R}} < N_{\boldsymbol{L}}$ and $N_{\boldsymbol{R}} < N_{\boldsymbol{A}}$. The sparsity pattern of $\boldsymbol{R}$ depends on the dropping tolerance and also the elements of the matrices $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$, but we are not going deeper here.

As discussed in Section 2.2 and Section 3.5, the sparse upper triangular matrix $\boldsymbol{R}$

is an incomplete Cholesky factor for the precision matrix $\boldsymbol{Q}$ of the GMRF when it is conditioned on data. The error matrix $\boldsymbol{E}$ between the true precision matrix $\boldsymbol{Q} = \boldsymbol{Q}_1 + \boldsymbol{Q}_2$ and the approximated precision matrix $\widetilde{\boldsymbol{Q}} = \boldsymbol{R}^{\mathrm{T}}\boldsymbol{R}$ is given by

$$\boldsymbol{E} = (\boldsymbol{Q}_1 + \boldsymbol{Q}_2) - \widetilde{\boldsymbol{Q}}. \tag{23}$$

The sparsity patterns of the precision matrix $\boldsymbol{Q}$ and its approximation $\widetilde{\boldsymbol{Q}}$ are shown in Figure 3(a) and Figure 3(b), respectively. In order to compare the difference between the approximated covariance matrices (inverse of the approximated precision matrix) $\widetilde{\boldsymbol{\Sigma}} = \widetilde{\boldsymbol{Q}}^{-1}$ and the true covariance matrix (inverse of the true precision matrix) $\boldsymbol{\Sigma} = \boldsymbol{Q}^{-1}$, we calculate the error matrix $\widetilde{\boldsymbol{E}}$,

$$\widetilde{\boldsymbol{E}} = \boldsymbol{\Sigma} - \widetilde{\boldsymbol{\Sigma}}. \tag{24}$$

The images of $\boldsymbol{\Sigma}$, $\widetilde{\boldsymbol{\Sigma}}$ and $\widetilde{\boldsymbol{E}}$ are given in Figure 4, and they show that the difference between $\boldsymbol{\Sigma}$ and $\widetilde{\boldsymbol{\Sigma}}$ is quite small. By chosen different dropping tolerance, the error can be made smaller and become negligible.
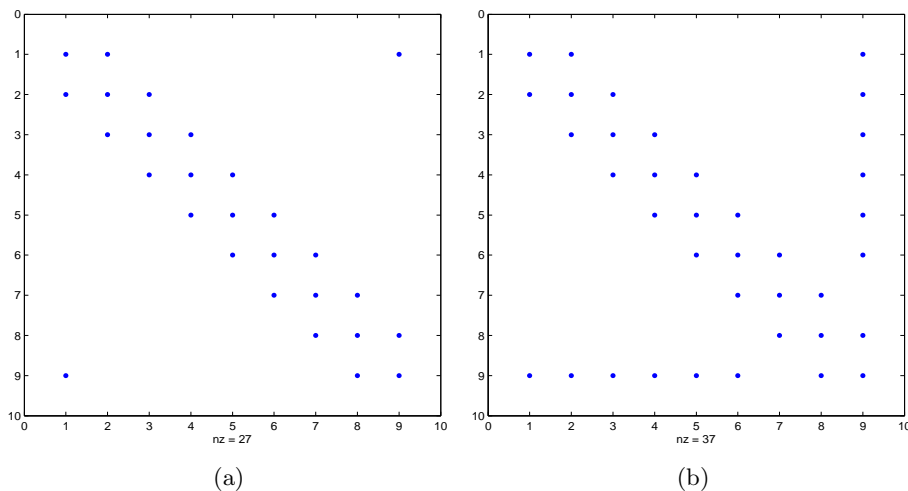
Figure 3: Sparsity patterns of the true precision matrix $\boldsymbol{Q}$ (a) and the approximated precision matrix $\widetilde{\boldsymbol{Q}}$ (b)

4 Simulation Results with cTIGO algorithm

Using the incomplete orthogonal factorization with Givens rotations, it leads to a sparse upper triangular matrix $\boldsymbol{R}$, which is a sparse incomplete Cholesky factor for the precision matrix $\boldsymbol{Q}$ and can be used to specify the GMRF. Hence it has the potential possibility to reduce the computational cost. We first apply the cTIGO algorithm to some commonly used structures of the precision matrices in Section 4.1. In Section 4.2, we apply the cTIGO algorithm to precision matrices which are generated from the stochastic partial differential equations (SPDEs) discussed in Lindgren et al. (2011) and Fuglstad (2011).
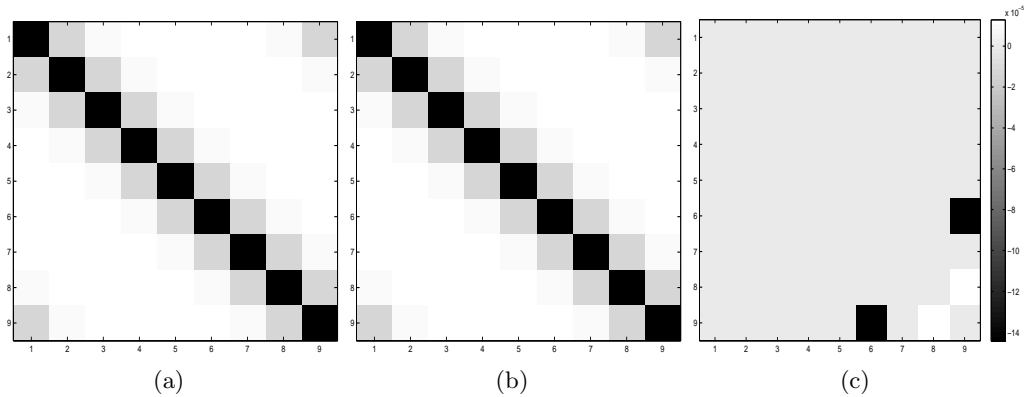
Figure 4: Images of the true covariance matrix $\boldsymbol{\Sigma}$ (a), approximately covariance matrix $\widetilde{\boldsymbol{\Sigma}}$ (b) and the error matrix $\widetilde{\boldsymbol{E}}$ (c)

4.1 Simulation results for precision matrices with commonly used structures

It is known that if the precision matrix $\boldsymbol{Q} > 0$ is a band matrix with bandwidth p, then its Cholesky factor $\boldsymbol{L}$ (lower triangular matrix) has the same bandwidth p. See Golub and Van Loan (1996) (Theorem 4.3.1) for a direct proof and Rue and Held (2005, Chapter 2.4.1) for more information on how to finding Cholesky factor efficiently in this case with Algorithm 2.9. Wist and Rue (2006) pointed out that if the original precision matrix $\boldsymbol{Q}$ is a band matrix, then the incomplete Cholesky factor $\widetilde{\boldsymbol{L}}$ from the incomplete Cholesky factorization will also be a band matrix with the same bandwidth p.

In this section we consider some commonly used structures for the precision matrices. The first two examples are band matrices with different bandwidths. Let x be Gaussian auto-regressive processes of order 1 or 2, and then the precision matrix for the process will be a band matrix with bandwidth $p = 2$ or $p = 3$, respectively. The precision matrices for the first-order Random Walk (RW1) and the second-order Random Walk (RW2) models have bandwidths $p = 2$ and $p = 3$. Since these models are intrinsic GMRFs, the precision matrices are not of full rank. We fix this by slightly modifying the elements in the precision matrices for the RW1 and RW2 models but we still called them as the precision matrices for the RW1 and the RW2 models. For more information about intrinsic GMRFs and the RW1 and RW2 models, see, for example, Rue and Held (2005, Chapter 3).

Assume that the data are Gaussian distributed. Then from Section 3.2 the matrix $\boldsymbol{Q}_2$ is a diagonal matrix when $\boldsymbol{A} = \boldsymbol{I}$. For simplicity and without lost of generality, assume the data $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, then the matrix $\boldsymbol{Q}_2$ and its Cholesky factor $\boldsymbol{L}_2$ are identity matrices. Since we know exactly what the sparsity patterns of the precision matrices $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ and the Cholesky factors $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ are, the sparsity pattern of $\boldsymbol{A}$ is known beforehand and can be taken advantage of in the implementation. By applying the cTIGO algorithm to the matrix $\boldsymbol{A}$ with dropping tolerance $\tau = 0.0001$, the sparse upper

18

triangular matrix $\boldsymbol{R}$ can be obtained. The sparsity patterns of the matrices $\boldsymbol{L}_1$, $\boldsymbol{L}_2$, $\boldsymbol{L}$, $\boldsymbol{A}$ and $\boldsymbol{R}$ are given in Figure 5. The sparsity patterns of the true precision matrix $\boldsymbol{Q}$ and the approximated precision matrix $\widetilde{\boldsymbol{Q}}$ are given in Figure 6. The image of the true covariance matrices $\boldsymbol{\Sigma}$, the approximated covariance matrix $\widetilde{\boldsymbol{\Sigma}}$ and the error matrix $\widetilde{\boldsymbol{E}}$ for the RW1 model are shown in Figure 7. Note that the order of the numerical values in the error matrix $\widetilde{\boldsymbol{E}}$ is 10^{-8}, which is essentially zero in practice applications.

Similarly for the RW2 model we apply the cTIGO algorithm to the matrix $\boldsymbol{A}$ with the dropping tolerance $\tau = 0.0001$. The results in this case are quite similar to the results for the RW1 model. We only show the images of the true covariance matrix $\boldsymbol{\Sigma}$, the approximated covariance matrix $\widetilde{\boldsymbol{\Sigma}}$, and the error matrix $\widetilde{\boldsymbol{E}}$. The results are given in Figure 8. Note that the order of the numerical values in the error matrix $\widetilde{\boldsymbol{E}}$ is 10^{-8} as for the RW1 model. See Section 4.4 from more simulation results for the RW1 and RW2 models and discussions. We can notice that the sparseness of $\boldsymbol{R}$ is the same as $\boldsymbol{L}$. Hence in these two cases, we do not save computational resources. However, this approach is still have the potential to be used in applications since it is robust.

The next example we have chosen is a block tridiagonal matrix of order n^2 resulting from discretizing Poisson's equation with the 5-point operator on an n-by-n mesh. Thus it is called Poisson matrix in this paper. The sparsity pattern of this matrix is given in Figure 10(a). With the Poisson matrix and $\boldsymbol{Q}_2$ as before, we find the Cholesky factors $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ and form the rectangular matrix $\boldsymbol{A}$. We apply the cTIGO algorithm to the matrix $\boldsymbol{A}$ with dropping tolerance $\tau = 0.0001$ to find the sparse upper triangular matrix $\boldsymbol{R}$. The sparsity patterns of the matrices $\boldsymbol{L}_1$, $\boldsymbol{L}_2$, $\boldsymbol{L}$, $\boldsymbol{A}$ and $\boldsymbol{R}$ are given in Figure 9(a) - Figure 9(e), respectively. The sparsity patterns of the true precision matrix $\boldsymbol{Q}$ and the approximated precision matrix $\widetilde{\boldsymbol{Q}}$ are given in Figure 10. We can notice that the upper triangular matrix $\boldsymbol{R}$ is sparser than the Cholesky factor $\boldsymbol{L}$ from the original precision matrix $\boldsymbol{Q}$. It can be shown that the sparseness depends on the dropping tolerance τ. The images of the true covariance matrices $\boldsymbol{\Sigma}$, the approximated precision matrix $\widetilde{\boldsymbol{\Sigma}}$, and the error matrix $\widetilde{\boldsymbol{E}}$ in this case are shown in Figure 11. Note that the order of the numerical values in the error matrix $\widetilde{\boldsymbol{E}}$ is 10^{-5}. This is small for practical use. More results for this band matrix are given in Section 4.4.

The next example is a precision matrix with a nearly band matrix. Assume that $\boldsymbol{Q}_1$ is a nearly banded matrix but with the values $\boldsymbol{Q}_1(1, n) = 1$ and $\boldsymbol{Q}_1(n, 1) = 1$. We call this matrix as Toeplitz matrix in this paper. The sparsity pattern of this matrix is given in Figure 13(a). With the dropping tolerance $\tau = 0.0001$, we apply the cTIGO algorithm to the rectangular matrix $\boldsymbol{A}$. The sparsity patterns of $\boldsymbol{L}_1$, $\boldsymbol{L}_2$, $\boldsymbol{L}$, $\boldsymbol{A}$ and $\boldsymbol{R}$ are given in Figure 12(a) - Figure 12(e), respectively. We notice that the upper triangular matrix $\boldsymbol{R}$ is sparser than the matrix $\boldsymbol{L}$. We can also notice that the sparseness of $\boldsymbol{R}$ depends on the tolerance τ. The sparsity pattern of the approximated precision matrix $\widetilde{\boldsymbol{Q}}$ is given in Fig 13(b). The image of the true covariance matrices $\boldsymbol{\Sigma}$, the approximated covariance matrix $\widetilde{\boldsymbol{\Sigma}}$, and the error matrix $\widetilde{\boldsymbol{E}}$ are shown in Figure 14. Note that the order of the numerical values in the error matrix is 10^{-5} with the given tolerance. More simulation results for this matrix are found in Section 4.4.
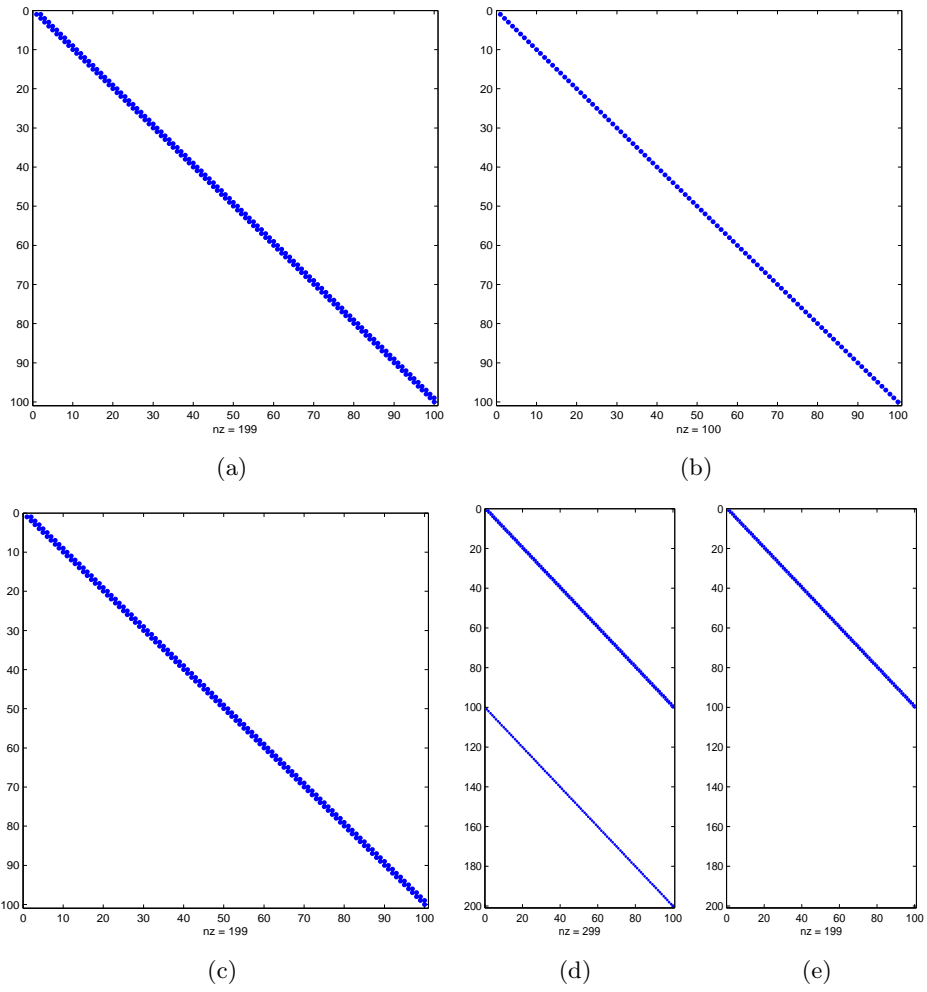
Figure 5: Sparsity patterns of $\boldsymbol{L}_1$ (a), $\boldsymbol{L}_2$ (b), $\boldsymbol{L}$ (c), $\boldsymbol{A}$ (d) and $\boldsymbol{R}$ (e) for the RW1 model

<table>
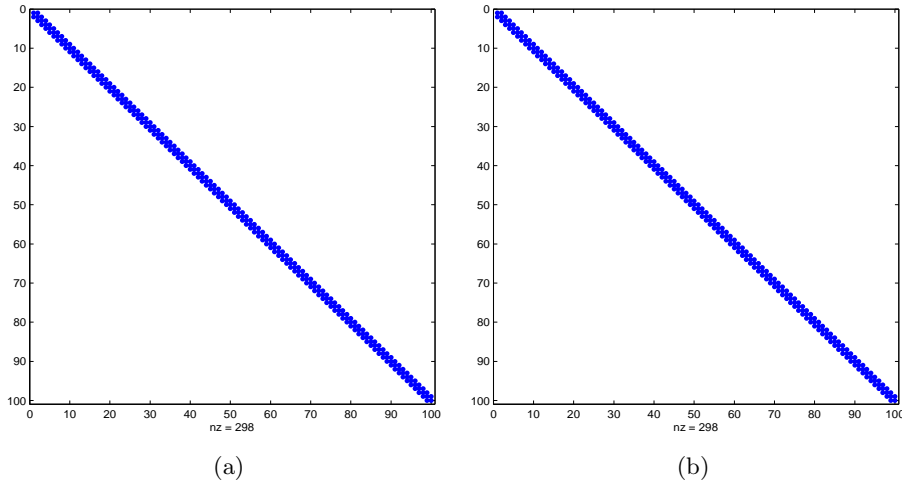<tr><td>(a)</td><td>(b)</td></tr>
</table>

Figure 6: Sparsity patterns for the true precision matrix $\boldsymbol{Q}$ (a) and the approximated precision matrix $\widetilde{\boldsymbol{Q}}$ (b) for the RW1 model

4.2 Simulation results for particular precision matrices

In this section we emphasize on some particular precision matrices, namely the precision matrices from the stochastic partial differential equations (SPDEs) approach discussed by Lindgren et al. (2011) and Fuglstad (2011). As pointed out by Lindgren et al. (2011) there is an explicit link between GRFs and GMRFs through SPDEs. The important relationship which was initially used by Lindgren et al. (2011) is that the solution $\boldsymbol{x}(\boldsymbol{u})$ to the following SPDE is a GRF with Matérn covariance function,

$$(\kappa^2 - \Delta)^{\alpha/2} x(\boldsymbol{s}) = \mathcal{W}(\boldsymbol{s}), \ \ \boldsymbol{s} \in \mathbb{R}^d, \ \ \alpha = \nu + d/2, \ \ \kappa > 0, \ \ \nu > 0, \tag{25}$$

where $\Delta = \sum_{i=1}^d \frac{\partial}{\partial x_i^2}$ is the Laplacian, $(\kappa^2 - \Delta)^{\alpha/2}$ is a differential operator and d is the dimension of the field $x(\boldsymbol{s})$. Fuglstad (2011) extended this approach to construct anisotropic and inhomogeneous fields with the SPDE

$$\kappa^2(\boldsymbol{u})x(\boldsymbol{u}) - \nabla \cdot \boldsymbol{H}(\boldsymbol{u})\nabla x(\boldsymbol{u}) = \mathcal{W}(\boldsymbol{u}), \tag{26}$$

where κ and $\boldsymbol{H}$ control the local range and anisotropy, and $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)$. One important difference between Lindgren et al. (2011) and Fuglstad (2011) is that Lindgren et al. (2011) have chosen the Neumann boundary condition but Fuglstad (2011) has chosen the periodic boundary condition. With Neumann boundary condition the precision matrix $\boldsymbol{Q}_1$ is a band matrix. However, the periodic boundary condition gives elements " in the corners" of the precision matrix. Hu et al. (2012) extended the approach to multivariate settings by using systems of SPDEs. For more information about the SPDE approach, We refer to Lindgren et al. (2011), Fuglstad (2011) and Hu et al. (2012).

First, choose the precision matrix for $\boldsymbol{Q}_1$ that results from the discretization of the SPDE (25) with $\alpha = 2$, $d = 2$ and $\kappa = 0.3$. The sparsity pattern of $\boldsymbol{Q}_1$ is given in Figure
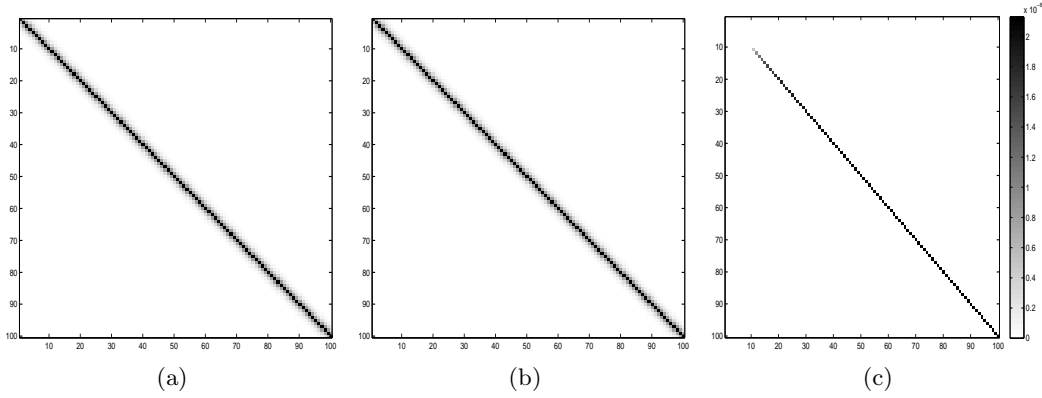
Figure 7: Images of true covariance matrix $\mathbf{\Sigma}$ (a), the approximated covariance matrix $\widetilde{\mathbf{\Sigma}}$ (b) and and the error matrix $\widetilde{\mathbf{E}}$ (c) for RW1 model
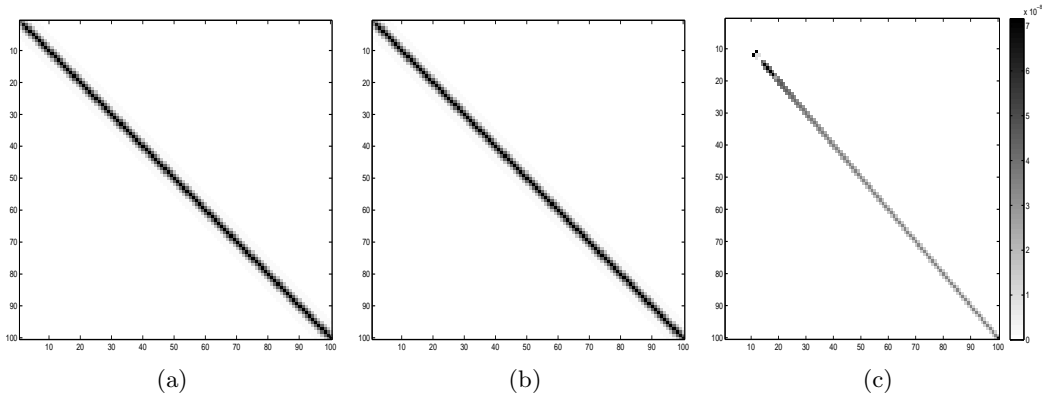
Figure 8: Images of the true covariance matrix $\mathbf{\Sigma}$ (a), the approximated covariance matrix $\widetilde{\mathbf{\Sigma}}$ (b) and and the error matrix $\widetilde{\mathbf{E}}$ (c) for RW2 model

16(a). We still assume $\mathbf{Q}_2 = \mathbf{I}$. The sparsity patterns of $\mathbf{L}_1$, $\mathbf{L}_2$, $\mathbf{L}$, $\mathbf{A}$ and $\mathbf{R}$ are given in Figure 15(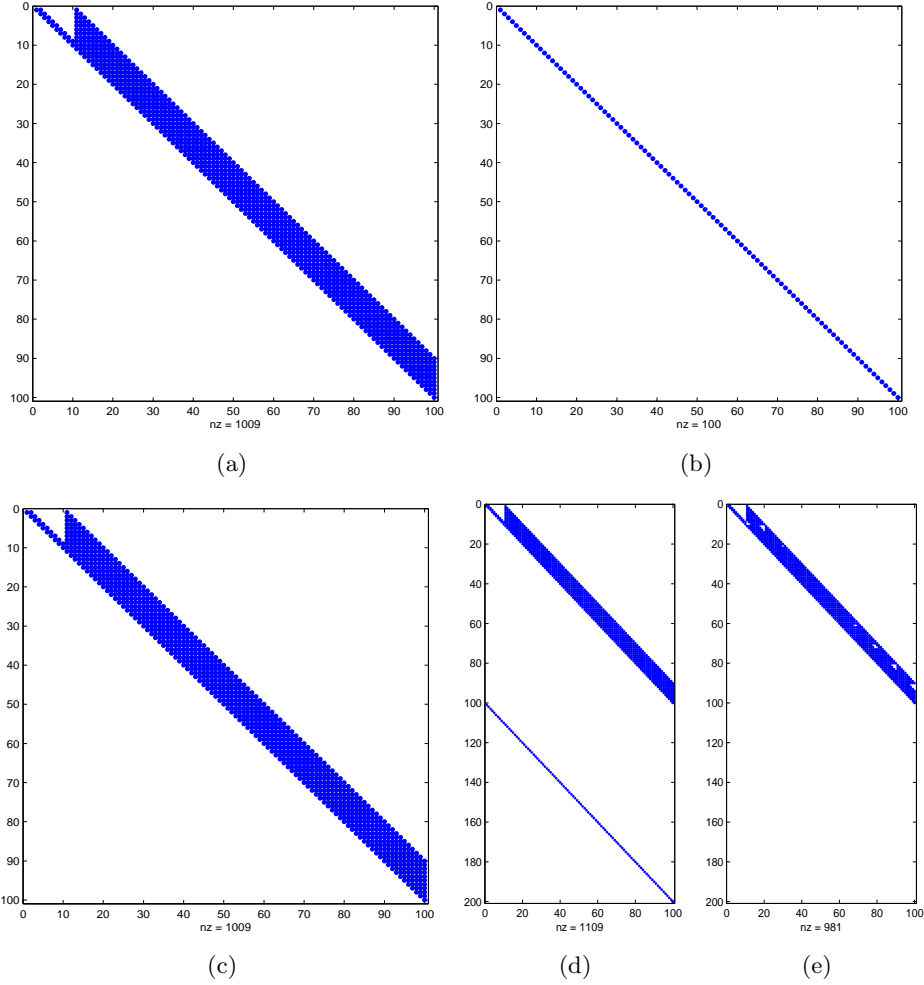a) -15(e), respectively. We notice that the upper triangular matrix $\mathbf{R}$ is sparser than the matrix $\mathbf{L}$. The sparsity pattern of the approximated precision matrix $\widetilde{\mathbf{Q}}$ is given in Figure 16(b). The images of the true covariance matrix $\mathbf{\Sigma}$, the approximated covariance matrix $\widetilde{\mathbf{\Sigma}}$ and the error matrix $\widetilde{\mathbf{E}}$ are shown in Figure 17. We can notice that the elements of in the error matrix $\widetilde{\mathbf{E}}$ are reasonably small.

The second precision matrix for $\mathbf{Q}_1$ is generated from the SPDE (26) with $\kappa = 0.1$ and

$$\mathbf{H} = 0.1 \times \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}.$$

The sparsity pattern of the precision matrix $\mathbf{Q}_1$ is given in Figure 19(a). We use the same $\mathbf{Q}_2$ as previous examples. The sparsity patterns of $\mathbf{L}_1$, $\mathbf{L}_2$, $\mathbf{L}$, $\mathbf{A}$ and $\mathbf{R}$ are given in Figure 18(a) - Figure 18(e), respectively. We can notice that the upper triangular

22

Figure 9: Sparsity patterns of $\boldsymbol{L}_1$ (a), $\boldsymbol{L}_2$ (b), $\boldsymbol{L}$ (c), $\boldsymbol{A}$ (d) and $\boldsymbol{R}$ (e) with Poisson matrix

matrix $\boldsymbol{R}$ is sparser than the matrix $\boldsymbol{L}$. The sparsity pattern of the approximated precision matrix $\widetilde{\boldsymbol{Q}}$ is given in Figure 19(b). The images of the true covariance matrix $\boldsymbol{\Sigma}$, the approximated covariance matrix $\widetilde{\boldsymbol{\Sigma}}$ and the error matrix $\widetilde{\boldsymbol{E}}$ are illustrated in Figure 20. We could notice that the order of the numerical values in the error matrix $\widetilde{\boldsymbol{E}}$ are also reasonably small in this case.

4.3 Sampling from GMRFs

In this section samples from a GMRF are obtained using the sparse upper triangular matrices $\boldsymbol{R}$ and the Cholesky factors $\boldsymbol{L}$ for the precision matrices $\boldsymbol{Q}$. Let the precision matrix $\boldsymbol{Q} = \boldsymbol{Q}_1 + \boldsymbol{Q}_2$, where $\boldsymbol{Q}_1$ is from the SPDE (25) or (26), and $\boldsymbol{Q}_2$ is a diagonal matrix. The sampling is done as follows.

23
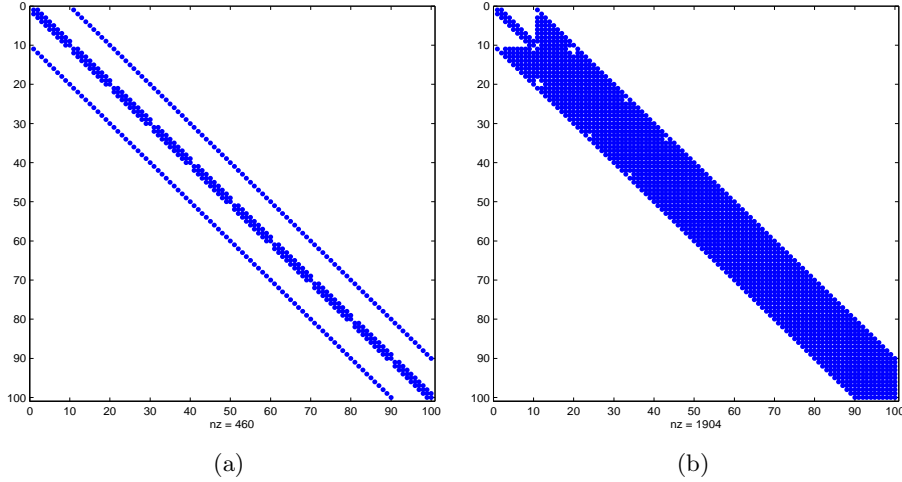
Figure 10: Sparsity patterns for the true precision matrix $\boldsymbol{Q}$ (a) and the approximated precision matrix $\widetilde{\boldsymbol{Q}}$ (b) with Poisson matrix

- Compute the Cholesky factor $\boldsymbol{L}$ with a Cholesky factorization or compute the sparse upper triangular matrix $\boldsymbol{R}$ from the cTIGO algorithm;

- Sample $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$;

- Solve the equation $\boldsymbol{L}^{\mathrm{T}} \boldsymbol{x} = \boldsymbol{z}$ or $\boldsymbol{R}\boldsymbol{x} = \boldsymbol{z}$;

- $\boldsymbol{x}$ is the sample of the GMRF with precision matrix $\boldsymbol{Q}$ or $\widetilde{\boldsymbol{Q}}$.

If the mean $\boldsymbol{\mu}$ of the field is not zero, then we just need a last step $\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{x}$ to correct the mean. With $\boldsymbol{L}$ the field $\boldsymbol{x}$ has the true covariance matrix $\boldsymbol{Q}$ because

$$\mathrm{Cov}(\boldsymbol{x}) = \mathrm{Cov}(\boldsymbol{L}^{-\boldsymbol{T}} \boldsymbol{z}) = (\boldsymbol{L}\boldsymbol{L}^{T})^{-1} = \boldsymbol{Q}^{-1}.$$

Similarly, with $\boldsymbol{R}$ the field $\boldsymbol{x}$ has the approximated covariance matrix $\tilde{\boldsymbol{Q}}$. Many other sampling algorithms are provided by Rue and Held (2005, Chapter 2) for different parametrization of the GMRF. We cannot notice any large differences between the samples using the Cholesky factor $\boldsymbol{L}$ and the samples using the sparse matrix $\boldsymbol{R}$ based on Figure 21 and Figure 22.

4.4 Effect of dropping tolerance

In this section we choose different values of τ in order to know the effect of dropping tolerance. We use the same kinds of structures for the precision matrices as discussed in Section 4.1 with dropping tolerances $\tau = \{0, 0.000001, 0.00001, 0.0001, 0.001, 0.01\}$. The 1-norm for the error matrix $\boldsymbol{E} = \boldsymbol{Q} - \widetilde{\boldsymbol{Q}}$ is used for the comparisons. The results are given in Table 1 and Table 2. From these tables, we can see that as the dropping tolerance τ becomes smaller and smaller, the error becomes smaller and smaller. Notice
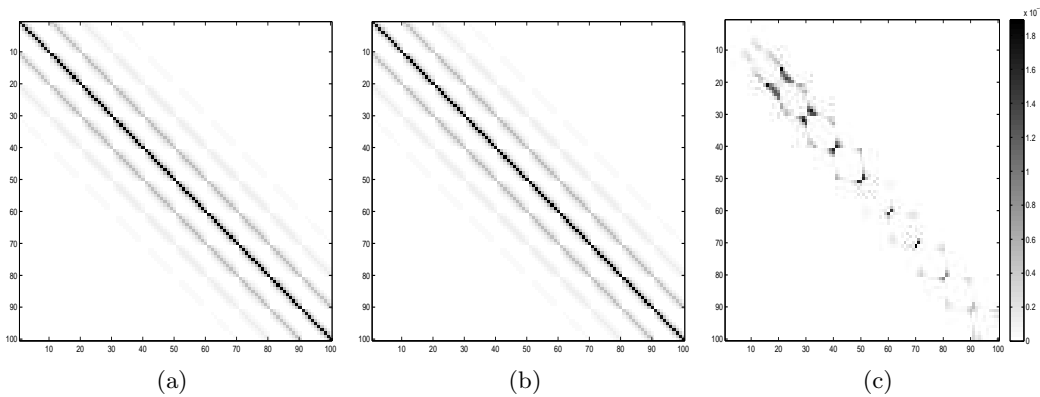
Figure 11: Images of true covariance matrix $\mathbf{\Sigma}$ (a), the approximated covariance matrix $\widetilde{\mathbf{\Sigma}}$ (b), and the error matrix $\widetilde{\mathbf{E}}$ (c) for Poisson matrix

that by choosing 10^{-5} as the dropping tolerance, the error reaches a level acceptable in many applications. If the dropping tolerance is equal to 0, then the error is also equal to zero which means no element has been zeroed out during the Givens rotations and it returns the common Cholesky factor.

Table 1: Comparisons for RW1 Model and RW2 Model

Random Walk 1		Random Walk 2	
tolerance	error	tolerance	error
0.01	2.55E-04	0.01	0.33
0.001	1.66E-06	0.001	1.80E-05
0.0001	2.13E-08	0.0001	1.48E-07
0.00001	2.50E-10	0.00001	1.07E-08
1.00E-6	2.34E-12	1.00E-6	2.15E-09
0	4.00E-15	0	1.73E-14

5   Discussion and Conclusion

In this paper we use the cTIGO algorithm to find sparse Cholesky factors for specifying GMRFs. Some commonly used structures of the precision matrices and two precision matrices generated from SPDEs have been tested. By using the incomplete orthogonal factorization with Givens rotations, a sparse incomplete Cholesky factor can be found and it is usually sparser than the Cholesky factor from the standard Cholesky factorization. The sparsity of the incomplete Cholesky factor depends on the value of the tolerance. With a good choice for the dropping tolerance, the error between the true covariance matrix and the approximated covariance matrix becomes negligible.

One advantage of this approach is that it is robust. It always produces a sparse incomplete Cholesky factor. Since the algorithm works both for square matrices and for
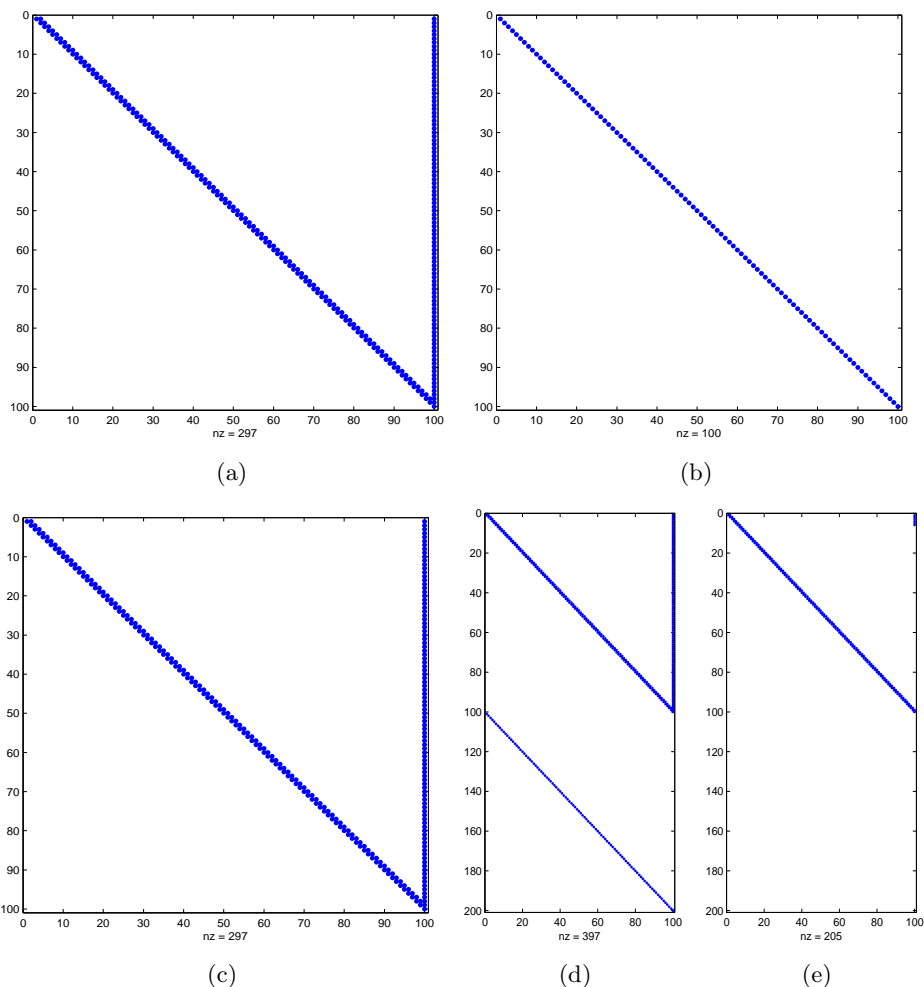
25

Figure 12: Sparsity patterns for $\boldsymbol{L}_1$ (a), $\boldsymbol{L}_2$ (b), $\boldsymbol{L}$ (c), $\boldsymbol{A}$ (d) and $\boldsymbol{R}$ for Toeplitz matrix.

rectangular matrices, this approach can be applied to GMRFs conditioned on observed data or a subset of the variable. On the negative side, it seems that our current implementation of the approach is slow when the dimension of the matrix becomes large. We believe that this is due to the nature of the incomplete orthogonal factorization with dynamic dropping strategy. The orthogonal factorization is usually slower than the Cholesky factorization. Further, Givens rotations only zero out values to zeros one at a time. This leads to the slowness of the algorithm. When the computation resources are limited, we might need to use the fixed pattern dropping strategy. However, to implement a fast cTIGO algorithm is out the scope of this paper and it is for further research.

(a)   (b)
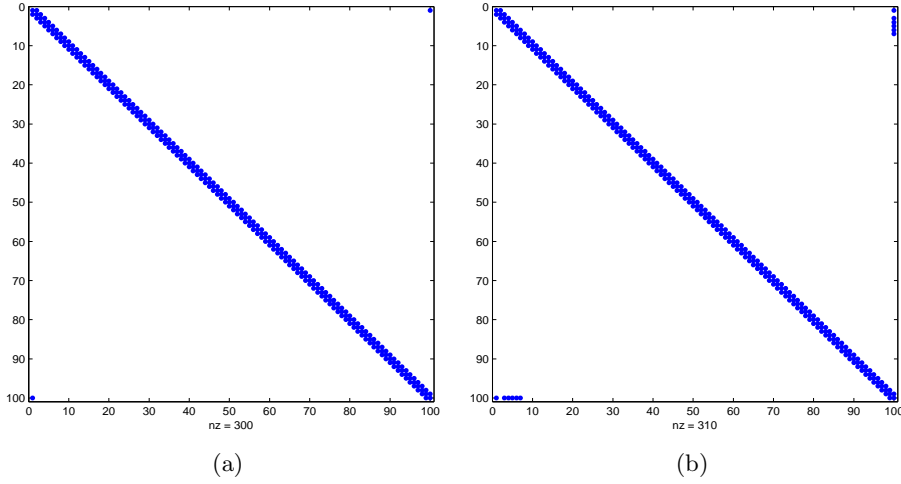
Figure 13: Sparsity patterns of $\boldsymbol{Q}$ (a) and $\boldsymbol{Qq}$ (b) for Toeplitz matrix.

Table 2: Comparisons for Poisson matrix and Toeplitz matrix

Poisson matrix		Toeplitz matrix	
tolerance	error	tolerance	error
0.01	0.11	0.01	9.15E-03
0.001	8.51E-03	0.001	9.59E-04
0.0001	6.33E-04	0.0001	7.91E-05
0.00001	5.91E-05	0.00001	8.70E-06
1.00E-6	3.44E-06	1.00E-6	7.19E-07
0	1.49E-14	0	5.66E-15

(a)   (b)   (c)
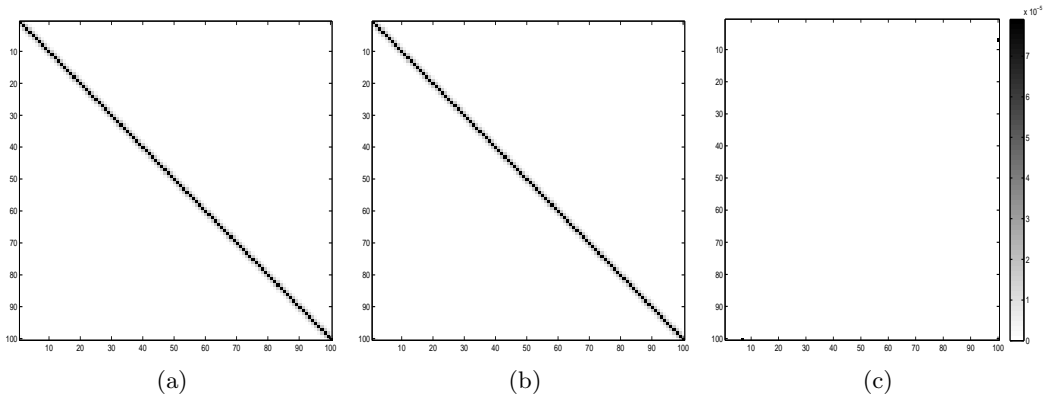
Figure 14: Images of the true covariance matrix $\boldsymbol{\Sigma}$ (a), the approximated covariance matrix $\widetilde{\boldsymbol{\Sigma}}$ (b) and the error matrix $\widetilde{\boldsymbol{E}}$ (c) for Toeplitz matrix.

27
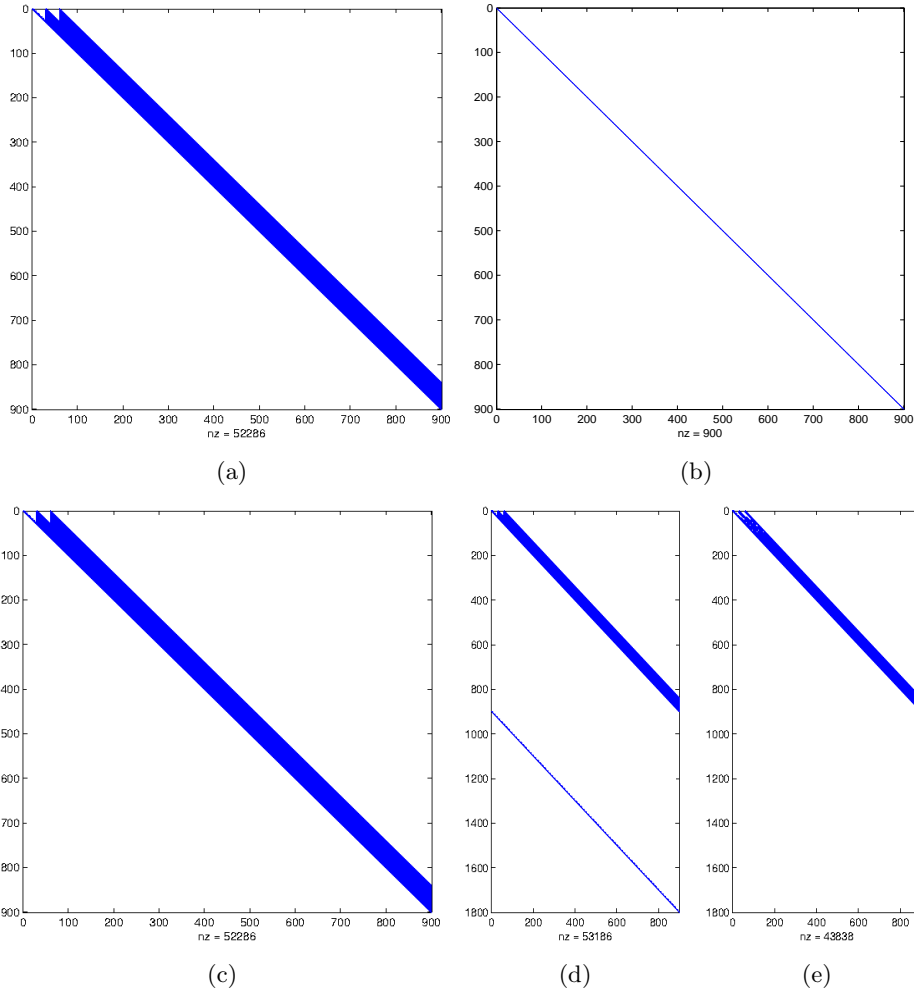
Figure 15: Sparsity pattern for $\boldsymbol{L}_1$(a), $\boldsymbol{L}_1$(b), $\boldsymbol{L}_1$(c), $\boldsymbol{A}$ (d) and $\boldsymbol{R}$(e) for the random field from the SPDE (25).
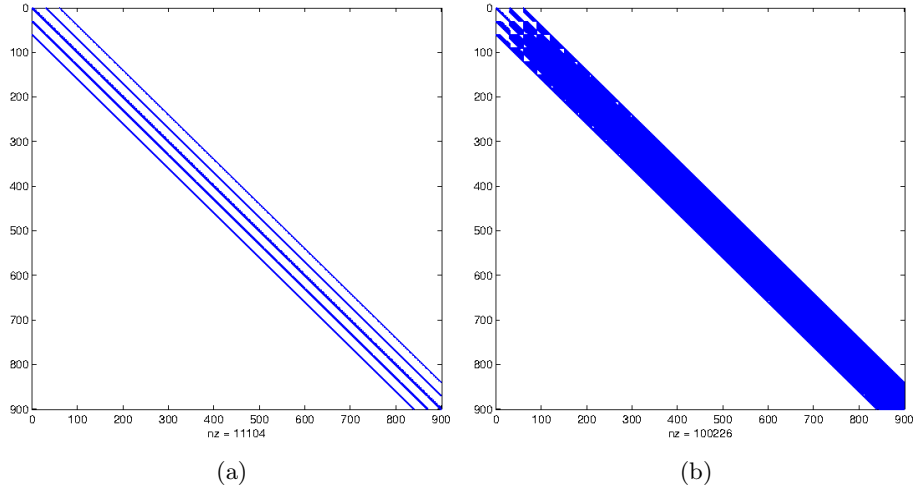
Figure 16: Sparsity patterns of $\boldsymbol{Q}$ (a) and $\boldsymbol{Qq}$ (b) for the random field generated from the SPDE (25).
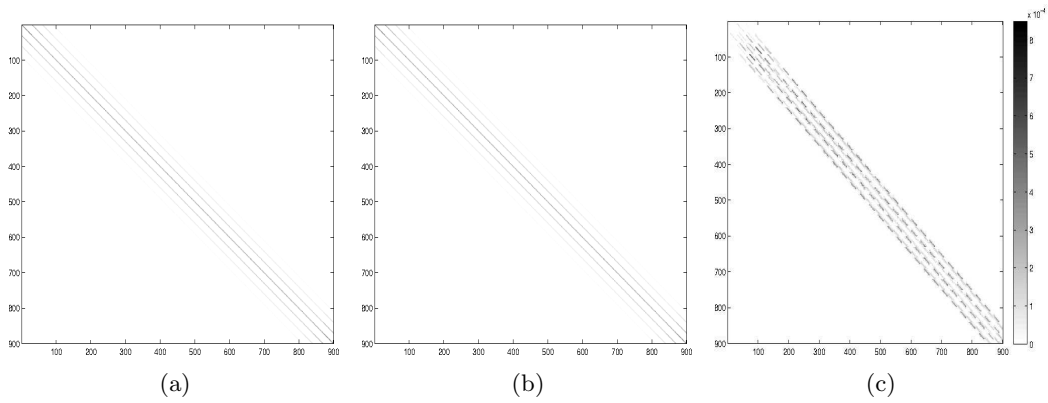
Figure 17: Images of the true covariance matrix $\boldsymbol{\Sigma}$ (a), the approxiamted covariance matrix $\widetilde{\boldsymbol{\Sigma}}$ (b) and the error matrix $\widetilde{\boldsymbol{E}}$ (c) for the random field generated from the SPDE (25).
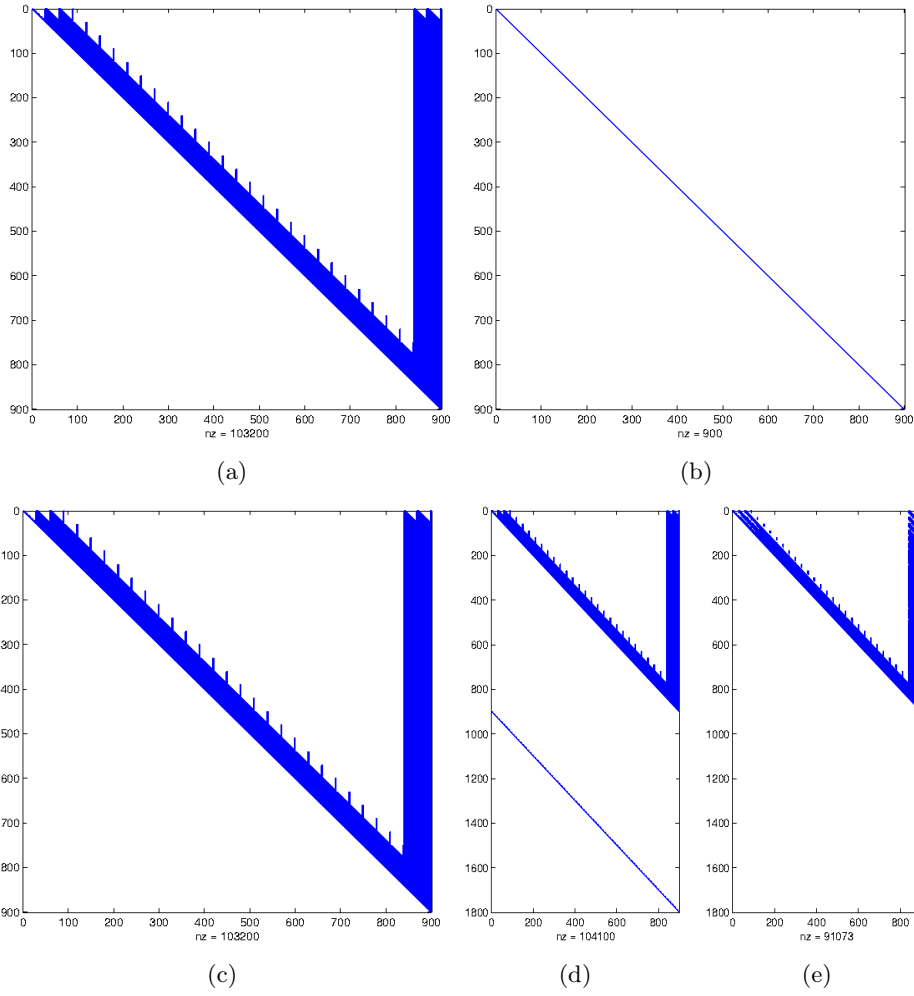
Figure 18: Sparsity pattern for $\boldsymbol{L}_1$(a), $\boldsymbol{L}_1$(b), $\boldsymbol{L}_1$(c), $\boldsymbol{A}$ (d) and $\boldsymbol{R}$(e) for the random field generated from SPDE given in (26).
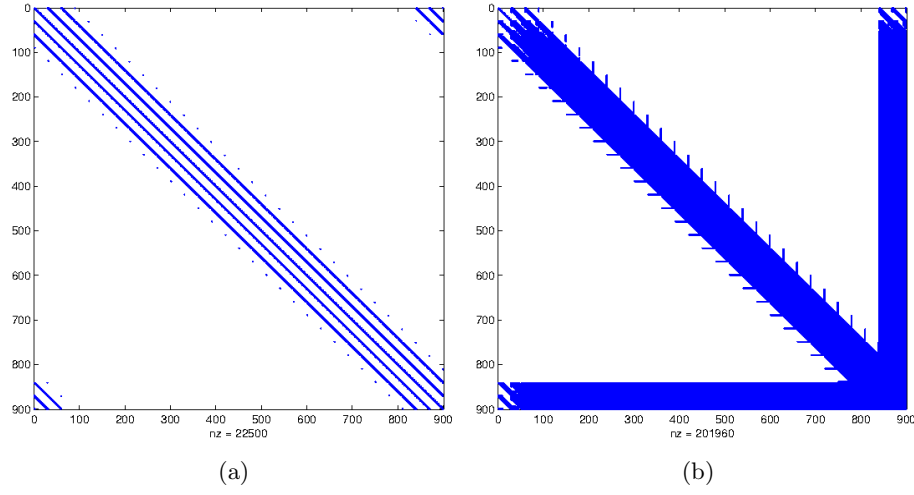
Figure 19: Sparsity patterns of $\boldsymbol{Q}$ (a) and $\boldsymbol{Qq}$ (b) for the random field generated from SPDE given in (26).
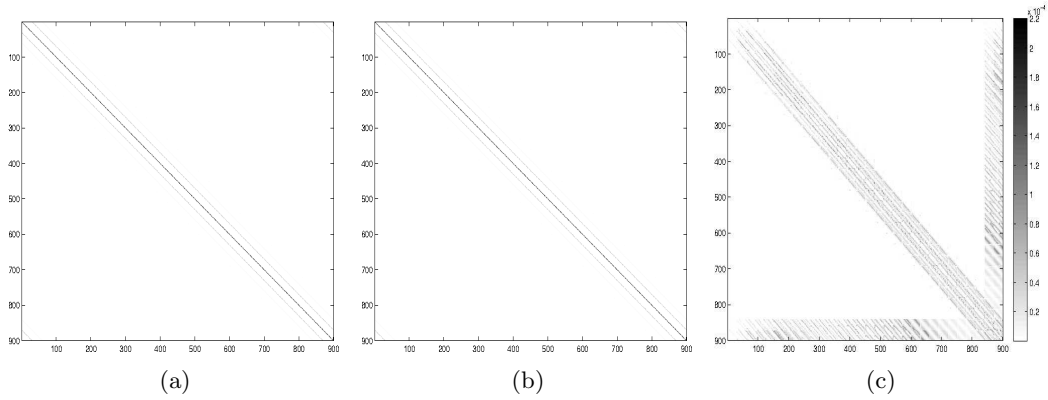
Figure 20: Images of the true covariance matrix $\boldsymbol{\Sigma}$ (a), the approxiamted covariance matrix $\widetilde{\boldsymbol{\Sigma}}$ (b) and the error matrix $\widetilde{\boldsymbol{E}}$ (c) for the random field generated from SPDE given in (26).

31
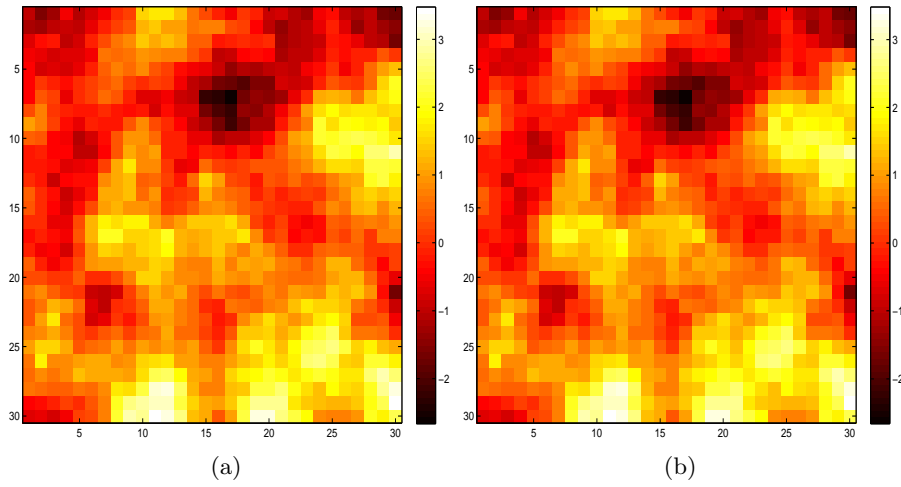
Figure 21: Samples from the GMRF with the common Cholesky factor $\boldsymbol{L}$ with (25) (a) and the upper triangular matrix $\boldsymbol{R}$ (b) from the cTIGO algorithm.
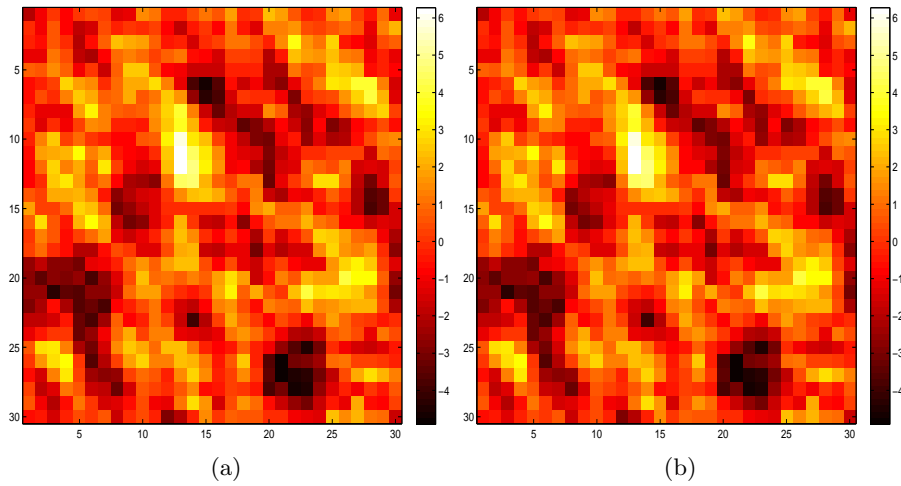

Figure 22: Samples from the GMRF with the common Cholesky factor $\boldsymbol{L}$ (a) with (26) and the upper triangular matrix $\boldsymbol{R}$ (b) from the cTIGO algorithm.

References

O. Axelsson. *Iterative solution methods.* Cambridge Univ Pr, 1996.

Z.Z. Bai and J.F. Yin. Modified incomplete orthogonal factorization methods using Givens rotations. *Computing*, 86(1):53–69, 2009. ISSN 0010-485X.

Z.Z. Bai, I.S. Duff, and A.J. Wathen. A class of incomplete orthogonal factorization methods. I: Methods and theories. *BIT Numerical Mathematics*, 41(1):53–70, 2001. ISSN 0006-3835.

Z.Z. Bai, I.S. Duff, and J.F. Yin. Numerical study on incomplete orthogonal factorization preconditioners. *Journal of Computational and Applied Mathematics*, 226(1):22–41, 2009. ISSN 0377-0427.

Å Björck. *Numerical methods for least squares problems.* Society for Industrial Mathematics, 1996. ISBN 0898713609.

Catherine Forbes, Merran Evans, Nicholas Hastings, and Brian Peacock. *Statistical distributions.* Wiley, 4 edition, 2011.

G.A. Fuglstad. Spatial modelling and inference with spde-based gmrfs. Master's thesis, Department of Mathematical Sciences, NTNU, 2011.

T. Gneiting, W. Kleiber, and M. Schlather. Matérn Cross-Covariance Functions for Multivariate Random Fields. *Journal of the American Statistical Association*, 105(491):1167–1177, 2010. ISSN 0162-1459.

G.H. Golub and C.F. Van Loan. *Matrix computations.* Johns Hopkins Univ Pr, 1996. ISBN 0801854148.

X. Hu, D.P. Simpson, F. Lindgren, and H. Rue. Multivariate gaussian random fields using systems of stochastic partial differential equations. *statistical report, Norwegian University of Science and Technology*, 2012.

A. Jennings and MA Ajiz. Incomplete Methods for Solving $A^T A x = b$. *SIAM Journal on Scientific and Statistical Computing*, 5:978, 1984.

F. Lindgren, H. Rue, and J. Lindström. An explicit link between gaussian fields and gaussian markov random fields: the stochastic partial differential equation approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(4):423–498, 2011.

J.A. Meijerink and H.A. van der Vorst. Guidelines for the usage of incomplete decompositions in solving sets of linear equations as they occur in practical problems. *Journal of computational physics*, 44(1):134–155, 1981.

N. Munksgaard. Solving sparse symmetric sets of linear equations by preconditioned conjugate gradients. *ACM Transactions on Mathematical Software (TOMS)*, 6(2):206–219, 1980. ISSN 0098-3500.

A.T. Papadopoulos, I.S. Duff, and A.J. Wathen. A class of incomplete orthogonal factorization methods. II: Implementation and results. *BIT Numerical Mathematics*, 45(1):159–179, 2005. ISSN 0006-3835.

H. Rue. Fast sampling of Gaussian Markov random fields. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):325–338, 2001. ISSN 1467-9868.

H. Rue. Marginal variances for Gaussian Markov random fields. *Statistics Report*, 2005.

H. Rue and L. Held. *Gaussian Markov random fields: theory and applications.* Chapman & Hall, 2005. ISBN 1584884320.

H. Rue, I. Steinsland, and S. Erland. Approximating hidden Gaussian Markov random fields. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(4):877–892, 2004. ISSN 1467-9868.

Y. Saad. Preconditioning techniques for nonsymmetric and indefinite linear systems* 1. *Journal of Computational and Applied Mathematics*, 24(1-2):89–105, 1988. ISSN 0377-0427.

Y. Saad. *Iterative methods for sparse linear systems*. Society for Industrial Mathematics, 2003. ISBN 0898715342.

D.P. Simpson. *Krylov subspace methods for approximating functions of symmetric positive definite matrices with applications to applied statistics and anomalous diffusion*. PhD thesis, Queensland University of Technology, 2008.

L.N. Trefethen and D. Bau. *Numerical linear algebra*. Society for Industrial Mathematics, 1997. ISBN 0898713617.

X. Wang, K.A. Gallivan, and R. Bramley. Cimgs: An incomplete orthogonal factorization preconditioner. *SIAM Journal on Scientific Computing*, 18(2):516–536, 1997.

H.T. Wist and H. Rue. Specifying a Gaussian Markov random field by a sparse Cholesky triangle. *Communications in Statistics-Simulation and Computation*, 35(1):161–176, 2006. ISSN 0361-0918.