SIAM J. SCI. COMPUT.
Vol. 34, No. 3, pp. B312–B334

SAMPLING GAUSSIAN DISTRIBUTIONS IN KRYLOV SPACES WITH CONJUGATE GRADIENTS*

ALBERT PARKER† AND COLIN FOX‡

Abstract. This paper introduces a conjugate gradient sampler that is a simple extension of the method of conjugate gradients (CG) for solving linear systems. The CG sampler iteratively generates samples from a Gaussian probability density, using either a symmetric positive definite covariance or precision matrix, whichever is more convenient to model. Similar to how the Lanczos method solves an eigenvalue problem, the CG sampler approximates the covariance or precision matrix in a small dimensional Krylov space. As with any iterative method, the CG sampler is efficient for high dimensional problems where forming the covariance or precision matrix is impractical, but operating by the matrix is feasible. In exact arithmetic, the sampler generates Gaussian samples with a realized covariance that converges to the covariance of interest. In finite precision, the sampler produces a Gaussian sample with a realized covariance that is the best approximation to the desired covariance in the smaller dimensional Krylov space. In this paper, an analysis of the sampler is given, and we give examples showing the usefulness and limitations of the Krylov approximations.

Key words. conjugate gradient, sampler, Gaussian, finite precision, Lanczos, eigenvalue, covariance, precision matrix, Krylov

AMS subject classifications. 65F10, 65F15, 62E17, 60G15, 60G60

DOI. 10.1137/110831404

1. Introduction.

Gaussian distributions are common throughout statistical modeling, being convenient from both computational and theoretical viewpoints. Not uncommonly the Gaussian model is defined on a state space with dimension 10^6 to 10^9 or more, in which case direct sampling algorithms that Cholesky factor the covariance matrix Σ can be very slow or infeasible. For example, high dimensional Gaussians are used in models of global total column ozone [13], tropical ocean surface winds [54], and the structure of the Earth's mantel and outer core [51]. High dimensional Gaussians also occur in exploratory analyses in inverse problems, for example, when the state corresponds to parameter values in a finite element discretization of a three-dimensional region. When working with Gaussian Markov random fields (GMRFs) [22, 38], the precision matrix Σ^{-1} is modeled. One advantage of this latter approach is that Σ^{-1} is sparse if the neighborhoods specifying conditional independence are small. Conventionally, efficient samplers must exploit the sparseness of either the covariance matrix Σ or the precision matrix Σ^{-1} within a Cholesky factorization to allow efficient sampling from the full Gaussian [37, 38], or use a circulant matrix structure that allows Fourier methods to be used [18].

The Cholesky factorization is also the preferred method for solving moderately sized linear systems when the coefficient matrix is symmetric and positive definite.

*Submitted to the journal's Computational Methods in Science and Engineering section April 18, 2011; accepted for publication (in revised form) March 23, 2012; published electronically June 26, 2012. This work was supported by the New Zealand Institute for Mathematics & its Applications thematic programme on PDEs.

http://www.siam.org/journals/sisc/34-3/83140.html

†Center for Biofilm Engineering, Montana State University, Bozeman, MT 59715 (parker@math.montana.edu).

‡Department of Physics, University of Otago, Dunedin 9016, New Zealand (fox@physics.otago.ac.nz).

For *large* linear systems, iterative solvers are the methods of choice due to their inexpensive cost per iteration and small computer memory requirements. For large dimensional Gaussians, iterative Gibbs sampling is one of the few general sampling methods available, often viewed as a reduction of the Metropolis–Hastings algorithm that takes advantage of the availability of conditional distributions. Perhaps less well known is that Gibbs samplers are essentially identical to stationary iterative methods [1, 2, 15, 21, 36] that were used as linear solvers in the 1950s and are now considered very slow due to their geometric rates of convergence.

An iterative Gaussian sampler with faster than geometric convergence was proposed by Schneider and Willsky in [44]. Just as the method of conjugate gradients (CG) determines search directions to the solution of $Ax = b$ in an optimal way rather than iterating through coordinate directions as does the stationary solver Gauss–Seidel, the sampler proposed by [44] uses conjugate directions to produce a sample, rather than the sequence of coordinate directions used by a Gibbs sampler which results in only geometric convergence [36].

In this paper, we expand upon the conjugate direction sampler of $N(0, A)$ proposed in [44], which was derived from the Lanczos algorithm for determining eigendecompositions and uses a stopping criterion based on matrix traces. We present a CG sampler, a simple extension of the CG linear solver, that, in addition to producing approximate samples from $N(0, A)$, also produces samples from $N(0, A^{-1})$ at no additional cost. This is relevant when modeling a GMRF, where $A := \Sigma^{-1}$ is a precision matrix model. This approach has the natural stopping criterion of terminating when CG finds a solution to $Ax = b$, that is, when the CG residual is equal to zero.

In exact arithmetic, CG is guaranteed to find a solution to the linear system $Ax = b$ after a finite number of iterations, and the Lanczos algorithm, for which CG is a special case [20, 26], finds all of the eigenpairs of A (when the eigenvalues of A are distinct). In finite precision, the search directions used by CG lose conjugacy. Nevertheless, CG still finds a solution to $Ax = b$ as long as "local conjugacy" of the search directions is maintained [26]. In fact, when the spectrum of A is clustered into k groups, CG finds the approximate solution after k iterations in a k-dimensional Krylov space. On the other hand, the loss of conjugacy of the search directions (which corresponds to loss of orthogonality of a Krylov basis) is detrimental to the Lanczos algorithm, so that only a few of the eigenvalues of A are estimated (the estimates are called *Ritz values*), the well-separated and extreme ones [26, 32, 40, 47]. The associated eigenvector estimates (*Ritz vectors*) are contained in the same k-dimensional Krylov space searched by CG.

The analysis in this paper shows that the CG sampler behaves like a Lanczos eigensolver in finite precision. Without corrective measures, loss of conjugacy prohibits sampling from the full Gaussian of interest, $N(0, A)$. The resulting sample has a realized covariance which is the best k-rank approximation to A (with respect to the 2-norm) in the k-dimensional Krylov space searched by the CG linear solver, and has Ritz pairs as eigenpairs. This result is different from that presented in [44], since, by driving the CG residual to zero, the CG sampler is guaranteed to converge to an invariant subspace of A. The CG sampler also produces approximate samples from $N(0, A^{-1})$, which have a realized covariance that is the best k-rank approximation to A^{-1} in the same Krylov space, and has reciprocal Ritz values and Ritz vectors as eigenpairs.

Similar to the difficulty faced by iterative eigenproblem solvers, the accuracy of the Krylov k-rank approximations of A and A^{-1} depends on the distribution of the eigenvalues of A. The accuracy of the covariance approximation of A can be quantified

by the fraction of mean-squared error reduction calculated via a ratio of matrix traces as the authors do in [44]. To determine how accurate the covariance approximation to A^{-1} is, we provide a similar inexpensive check using a CG implementation of the Monte Carlo Lanczos scheme outlined in [5].

CG has the remarkable property of solving $Ax = b$ in a finite number of steps while requiring storage of only two vectors and the ability to operate by the matrix A. In exact arithmetic, the CG sampler has the analogous property of producing samples from the Gaussians $N(0, A)$ and $N(0, A^{-1})$ in a finite number of steps while requiring storage of a single state vector in addition to those required by CG. Within the sampler one needs to operate by the matrix A, but there is no need to store the matrix or factorize it. Hence the sampler is useful in high dimensional problems where forming A is impractical or inconvenient, and it gives computational efficiency in those problems where operation by A can be performed much more cheaply than by direct matrix multiplication.

In finite precision, however, to attain a specified level of accuracy of a covariance approximation, one must correct for the loss of conjugacy of the search directions at the expense of more computational time and/or increased memory requirements. This is the same conundrum faced by any Lanczos method as the number of eigenpairs of A that one wishes to estimate increases. The sampler proposed by Schneider and Willsky [43, 44] uses reorthogonalization schemes [20, 32], commonly used with Lanczos processes. Other approaches such as spectral transformations [16, 27, 48, 56], block Lanczos methods [4, 6, 20, 28, 32, 56]), and *restarting* and *look-ahead* schemes [4, 48] have also been used successfully with Lanczos methods. For high dimensional problems, these schemes can be as expensive as a Cholesky factorization (depending on the distribution of the eigenvalues) and hence are infeasible.

When drawing samples from large-scale Gaussians, other corrective measures need to be investigated. One obvious way to generate a sample with a realized covariance arbitrarily close to A or A^{-1}, while storing only vectors (instead of a matrix as large as A), is to use a Gibbs sampler, accelerated by initialization with a CG sample. Unfortunately, the geometric convergence of the Gibbs sampler is reduced by only a constant factor in this case. A more appealing approach is suggested by the result in this paper that the realized covariance of the CG samples is equal to the CG polynomial described in [3, 30, 31]. Thus, the same polynomial filters used to accelerate linear solvers [19, 31, 39, 41, 48] can potentially be used to inexpensively aid conjugate direction samplers by damping out the eigenspaces that have already been sampled. The key difference between Lanczos eigensolvers and Krylov samplers is that samplers need not explicitly estimate eigenpairs in order to sample from the corresponding eigenspaces. This is an area of current research.

The rest of the paper is organized as follows. In section 2 we present the CG sampler, determine the distribution of the samples, and give some necessary background of the Lanczos algorithm. Section 3 is the central part of the paper. We make clear the relationship between CG sampling and Lanczos and show that the realized covariance matrices of the CG samples after k iterations are accurate in the associated Krylov spaces. We also provide inexpensive checks, based on matrix traces, to provide some measure of the accuracy of the realized covariances. Last, we consider CG sampling from a Gaussian with a symmetric positive semidefinite covariance or precision matrix with a nontrivial nullspace. In section 4 we present some numerical examples from commonly used Gaussian models which demonstrate how the sampler behaves in finite precision, and we compare the results to those from Cholesky and Gibbs samplers.

2. Mathematical preliminaries.

2.1. Sampling with conjugate gradients. An n-dimensional Gaussian with zero mean is defined by its $n \times n$ symmetric and positive definite covariance matrix Σ. We denote the distribution by $N(0, \Sigma)$. Throughout what follows, we allow that either $A := \Sigma$ (in which case $N(0, A)$ is of interest), or $A := \Sigma^{-1}$ (in which case $N(0, A^{-1})$ is of interest).

The conventional way to sample from a Gaussian with a given covariance matrix A is to determine the Cholesky factorization $A = CC^T$, so that if $z \sim N(0, I)$, then $c = Cz \sim N(0, A)$. If A is a precision matrix, then solving $C^T y = z$ gives $y = C^{-T} z \sim N(0, A^{-1})$ [37, 38]. The Cholesky factorization is the method of choice since it is fast, incurring approximately $1/3n^3$ floating point operations (flops) [20, p. 144], [53, p. 40], and is backwards stable [53]. If A has bandwidth b, calculation of the Cholesky factorization requires $\mathcal{O}(b^2 n)$ flops, which is a substantial savings when $b \ll n/2$ [20, 37, 53].

In addition to the Cholesky factorization, other linear solvers have been used to sample from Gaussian distributions. The correspondence between linear solvers and Gaussian samplers is due to the fact that a linear solve of $Ax = b$ is the same as minimizing the quadratic

$$\phi(x) = \frac{1}{2} x^T A x - b^T x,$$

and $\exp(-\phi)$ is proportional to the Gaussian density $N(A^{-1}b, A^{-1})$. For example, this relationship is used to show the equivalence between Gauss–Seidel, which iteratively solves a linear system by minimizing ϕ in (blocks of) coordinate directions, and the Gibbs sampler, which samples from the Gaussian $N(0, A^{-1})$ conditional on blocks of the coordinate random variables [1, 2, 36].

Iterative samplers such as a Gibbs sampler are attractive options when sampling from high dimensional Gaussians due to their inexpensive cost per iteration (about $2n^2$ flops) and small computer memory requirements (only vectors of size n need be stored). If the precision matrix is sparse with $\mathcal{O}(n)$ nonzero elements, then, regardless of the bandwidth, iterative methods cost only about $2n$ flops per iteration, which is competitive with sparse Cholesky factorizations (i.e., if each row of A has about $s \ll n$ nonzero elements, then the cost of an iteration, dominated by the matrix-vector multiply, is about $2sn$ flops). Unfortunately, the current state of the art for iterative samplers, which is equivalent to symmetric successive overrelaxation (SSOR), converges only geometrically [36].

An iterative sampler of $N(0, A)$ that is guaranteed to converge in a finite number of steps (in exact arithmetic) was proposed by Schneider and Willsky in [44]. This method uses a Lanczos process to produce conjugate directions and then generates samples along these directions, at a cost of $\mathcal{O}(n^2)$ flops per iteration.

We propose the following algorithm to produce samples $y \sim N(0, A^{-1})$ and $c \sim N(0, A)$. Instead of using a Lanczos eigensolver to generate conjugate directions, this sampler uses the CG method to solve the linear system $Ax = b$.

ALGORITHM 1 (CG sampler from $N(0, A^{-1})$). *Given $n \times 1$ vectors b and x^0, and an $n \times n$ symmetric positive definite matrix A, let $r^0 = b - Ax^0$, $p^0 = r^0$, $d_0 = p^{(0)T} A p^0$, $y^0 = x^0$, and $k := 1$. Specify some stopping tolerance ϵ. Iterate:*

1. $\gamma_{k-1} = \frac{r^{(k-1)T} r^{k-1}}{d_{k-1}}$ *is the one-dimensional minimzer of ϕ in the direction $x^{k-1} + \gamma p^{k-1}$.*
2. $x^k = x^{k-1} + \gamma_{k-1} p^{k-1}$.

3. *Sample* $z \sim N(0,1)$*, and set* $y^k = y^{k-1} + \frac{z}{\sqrt{d_{k-1}}} p^{k-1}$.
4. $r^k = -\nabla_x \phi(x^k) = r^{k-1} - \gamma_{k-1} A p^{k-1}$ *is the residual.*
5. $\beta_k = -\frac{r^{kT} r^k}{r^{(k-1)T} r^{k-1}}$.
6. $p^k = r^k - \beta_k p^{k-1}$ *is the next conjugate search direction.*
7. $d_k = p^{(k)T} A p^k$.
8. *Quit if* $||r^k||_2 < \epsilon$*. Else set* $k := k+1$ *and go to step* 1.

Note that implementing only steps 1–2 and 4–8 is the standard CG algorithm for solving $Ax = b$. The addition of step 3 which produces the Gaussian sample y^k incurs a negligible additional cost of a vector addition and store at each iteration, so that the total cost of Algorithm 1 is the same as that of CG, $2n^2$ flops per iteration. In section 3.5 we consider implementation of Algorithm 1 when A is symmetric and positive semidefinite with a nontrivial nullspace.

We will show in section 3.1 that after k iterations, y^k has a Gaussian distribution with a covariance which approximates A^{-1},

$$y^k \dot{\sim} N(0, A^{-1})$$

(which is accurate as long as the eigenspaces contained by $\text{span}(\{p^i\})$ correspond to the small eigenvalues of A). This suggests that

$$c^k = Ay^k \dot{\sim} N(0, A)$$

(which is accurate when the eigenspaces contained by $\text{span}(\{p^i\})$ correspond to the large eigenvalues of A). In fact, replacing step 3 in Algorithm 1 with

$$\text{3. Sample } \; z \sim N(0,1), \text{ and set } c^k = c^{k-1} + \frac{z}{\sqrt{d_{k-1}}} A p^{k-1}$$

is equivalent (in exact arithmetic) to how the samples are generated using a Lanczos method in [44]. However, instead of step 8, the authors in [44] terminate the conjugate direction sampler when $\text{trace}\big(\text{Var}(c^k)\big) \approx \text{trace}(A)$. This difference in a stopping criterion assures the optimality of the realized covariances of the CG samples y^k and c^k in $\text{span}(\{p^i\})$ (see section 3.1).

2.2. Distributions of the CG samples. In order to show that the distributions of the CG samples y^k and c^k produced by Algorithm 1 converge to the Gaussians of interest, we first need to establish some notation. Let P_k be the $n \times k$ matrix with the search directions $\{p^i\}_{i=0}^{k-1}$ as columns. The k vectors $\{p^i\}_{i=0}^{k-1}$ are *A-conjugate* ($p^{iT} A p^j = 0$ for $i \neq j$), and the residuals $\{r^i\}_{i=0}^{k-1}$, where $r^i = b - Ax^i$, are orthogonal. The span of each of these sets is equal to the *Krylov space* of dimension k [30],

$$\mathcal{K}^k(A, r^0) := \text{span}(r^0, Ar^0, A^2 r^0, \ldots, A^{k-1} r^0).$$

Let $\widetilde{P}_k$ be the $n \times (n-k)$ matrix whose columns are the conjugate directions $\{p^i\}_{i=k}^{n-1}$. Then $P_n = [P_k \;\; \widetilde{P}_k]$ is invertible, and

$$D_n := \begin{pmatrix} D_k & 0 \\ 0 & \widetilde{D}_k \end{pmatrix} = \begin{pmatrix} P_k^T A P_k & 0 \\ 0 & \widetilde{P}_k^T A \widetilde{P}_k \end{pmatrix} = P_n^T A P_n$$

is an invertible diagonal matrix, $[D_n]_{ii} := p^{iT} A p^i$. Thus,

$$A^{-1} = P_n D_n^{-1} P_n^T = P_k D_k^{-1} P_k^T + \widetilde{P}_k \widetilde{D}_k^{-1} \widetilde{P}_k^T.$$

Now the CG sample y^k can be written as $y^k = y^0 + P_k D_k^{-1/2} z^k$, where $z^k \sim N(0, I_k)$. Thus, when the CG sampler terminates after $k < n$ iterations, the mean and covariance are

$$\begin{aligned} \mathrm{E}(y^k|y^0, b) &= y^0, \\ \mathrm{Var}(y^k|y^0, b) &= \mathrm{Var}(P_k D_k^{-\frac{1}{2}} z^k) = P_k D_k^{-1} P_k^T. \end{aligned} \tag{2.1}$$

If either y^0 or b is random, then the unconditional mean and covariance are [12]

$$\begin{aligned} \mathrm{E}(y^k) &= \mathrm{E}(\mathrm{E}(y^k|y^0, b)) = \mathrm{E}(y^0), \\ \mathrm{Var}(y^k) &= \mathrm{E}\left(\mathrm{Var}(y^k|y^0, b)\right) + \mathrm{Var}\left(\mathrm{E}(y^k|y^0, b)\right) \\ &= \mathrm{E}(P_k D_k^{-1} P_k^T) + \mathrm{Var}(y^0). \end{aligned} \tag{2.2}$$

Thus, setting y^0 to a random vector introduces an additional component to $\mathrm{Var}(y^k)$. If the CG sampler is initialized with $y^0 = x^0 = 0$, then (since $c^k = Ay^k$)

$$y^k|b \;\sim\; N(0, P_k D_k^{-1} P_k^T) \qquad \text{and} \qquad c^k|b \;\sim\; N(0, A P_k D_k^{-1} P_k^T A).$$

Since the covariance matrix $P_k D_k^{-1} P_k^T$ is singular, the conditional distributions of y^k and c^k are called *intrinsic Gaussians* in [38].

In exact arithmetic at iteration $k = n$, as long as A has n distinct eigenvalues, the CG sampler produces samples

$$y^n \;\sim\; N(0, A^{-1}) \qquad \text{and} \qquad c^n \;\sim\; N(0, A).$$

When A does not have n distinct eigenvalues, CG terminates at iteration $k < n$ [26]. Just as convergence of Lanczos can be aided by preconditioning to spread out the eigenvalues of A, (e.g., when A has repeated eigenvalues), one can precondition the CG sampler by $\tilde{A} = U^T A U$ for some invertible U so that $\tilde{A}$ has distinct eigenvalues. Now the sampler produces $\tilde{y} \sim N(0, \tilde{A}^{-1})$ and $\tilde{c} = \tilde{A}\tilde{y} \sim N(0, \tilde{A})$ and hence $U\tilde{y} \sim N(0, A^{-1})$ and $U^{-T}\tilde{c} \sim N(0, A)$. As with a preconditioned CG (PCG) linear solver, a PCG sampler does not require computation of either U or its inverse, but only requires multiplication of UU^T at each iteration [30, 41].

2.3. The Lanczos method for estimating eigenpairs. The behavior of CG in finite precision has been described by considering CG as a Lanczos process [3, 11, 14, 20, 26]. We will present the Lanczos algorithm in this section, with the goal of describing the accuracy of the distributions of the CG samples in section 3.

The Lanczos algorithm is an iterative method for estimating the eigenpairs (λ_i, w^i) of a given $n \times n$ positive definite matrix A [23, 24], and its performance in finite precision is well studied [26, 32, 40]. At the kth iteration, the Lanczos algorithm is equivalent to the matrix equation

$$AV_k = V_k T_k + \eta_k v^k e^{kT}, \tag{2.3}$$

where e^i is the ith column of the $k \times k$ identity I_k, the Lanczos vector v^{i-1} is the ith column of the $n \times k$ matrix V_k, and the $k \times k$ tridiagonal Lanczos matrix is

$$T_k := \begin{pmatrix} \alpha_0 & \eta_1 & & & \\ \eta_1 & \alpha_1 & \eta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \eta_{k-2} & \alpha_{k-2} & \eta_{k-1} \\ & & & \eta_{k-1} & \alpha_{k-1} \end{pmatrix}.$$

The eigenvalues of T_k, which we will index by either

$$\theta_1^k < \cdots < \theta_k^k \quad \text{or} \quad \theta_{-k}^k < \cdots < \theta_{-1}^k,$$

are called *Ritz values* [11, 26, 32] and are the Lanczos estimates of k of the eigenvalues $\{\lambda_i\}$ of A. Letting $q^1, \ldots, q^k$ denote the orthonormal eigenvectors of T_k, the *Ritz vectors* $\{V_k q^i\}_{i=1}^k$ are the Lanczos estimates of the k eigenvectors $\{w^i\}$ of A corresponding to $\{\lambda_i\}$. Thus, to estimate the eigenvectors of A, Lanczos must either store the potentially huge $n \times k$ matrix V_k or, for each eigenvalue estimate, apply inverse iteration [4]. The spectral decomposition of T_k can now be written as

$$T_k = Q_k \Theta_k Q_k^T, \tag{2.4}$$

where $\Theta = \text{diag}(\theta_1, \ldots, \theta_k)$ and Q_k has the eigenvectors q^i as columns.

Multiplying (2.3) on the right by the eigenvector q^i shows that

$$||AV_k q^i - \theta_i^k V_k q^i||_2 = \eta_k |q_k^i|, \tag{2.5}$$

where q_k^i is the last component of q^i. This shows that $\eta_k |q_k^i| = 0$ signals convergence of $(\theta_i^k, V_k q^i)$ to (λ_i, w^i) [26, p. 9], [32, p. 260]. In particular, $\eta_k = 0$ indicates that all of the Ritz pairs have converged.

By construction, in exact arithmetic, the k Lanczos vectors are an orthonormal basis for $\mathcal{K}^k(A, v^0)$ [20, 26]. We already mentioned in section 2.1 that the normalized CG residual vectors $r^0/||r^0||_2, \ldots, r^{k-1}/||r^{k-1}||_2$ form an orthonormal basis for $\mathcal{K}^k(A, r^0)$ for any k. Thus, if the Lanczos algorithm is initialized with $v^0 = r^0/||r^0||_2$, then, up to a sign change, the Lanczos vectors are normalized CG residuals, and, in fact [14, 20, 26],

$$v^k = (-1)^k \frac{r^k}{||r^k||}. \tag{2.6}$$

This is the key relationship between CG and Lanczos that we will exploit. The consequences which we will use repeatedly are summarized in the next lemma.

LEMMA 2.1.

1. *A QR factorization of the matrix of conjugate directions P_k is $P_k = V_k R_k$, where V_k is the matrix of orthonormal Lanczos vectors and $R_k = V_k^T P_k$ is upper triangular.*
2. *$T_k^{-1} = R_k D_k^{-1} R_k^T$ and $P_k D_k^{-1} P_k^T = V_k T_k^{-1} V_k^T$.*
3. *The nonzero elements of T_k, $\{\alpha_i\}$ and $\{\eta_i\}$, can be calculated from the CG parameters $\{\gamma_i\}$ and $\{\beta_i\}$ by*

$$\alpha_i = \frac{1}{\gamma_i} - \frac{\beta_i}{\gamma_{i-1}} = \frac{d_i}{||r^i||_2^2} - \frac{d_{i-1}||r^i||_2^2}{||r^{i-1}||_2^4}, \qquad \eta_{i+1} = \frac{\sqrt{-\beta_{i+1}}}{\gamma_i} = \frac{d_i ||r^{i+1}||_2}{||r^i||_2^3},$$

   where $0 \le i < k-1$ and the convention $\beta_0 = 0$ and $\gamma_{-1} = 1$ is used (this well-known result can be found in [14, 26, 42]*).*
4. *The CG one-dimensional minimizer γ_{k-1} is a function of the eigenpairs (θ_i^k, q^i) of the $k \times k$ Lanczos tridiagonal T_k,*

$$\sum_{i=1}^k \frac{(q_k^i)^2}{\theta_i^k} = e^{kT} T_k^{-1} e^k = \gamma_{k-1}.$$

Proof. See Appendix A.1. ∎

Parts 1, 2, and 4 of Lemma 2.1 will allow us to show that the CG covariance matrices are optimal in the Krylov space $\mathcal{K}^k(A, r^0)$, and part 3 shows how CG can be used to construct the $k \times k$ Lanczos tridiagonal matrix T_k and then inexpensively estimate some of the eigenvalues of A.

REMARK 2. *In exact arithmetic, comparing* (2.3) *and* (2.5) *with Lemma* 2.1.3 *shows that the CG residual* $r^k = 0$ *corresponds to CG finding a solution to* $Ax = b$ *at the same time that the CG sampler finds an invariant subspace of* A *when all* k *Ritz pairs* $(\theta_i^k, V_k y^i)$ *converge to* k *of the eigenpairs of* A *(see also* [20, p. 492], [26, p. 50]*). On the other hand, this also shows that some Ritz pairs can converge first (as in one of the examples presented in section* 4*).*

2.4. Lanczos in finite precision. CG is remarkably robust in finite precision. In fact, if the eigenvalues of A are in k distinct clusters, then CG tends to find an approximate solution to $Ax = b$ after only k iterations [30, 47]. As long as "local conjugacy" is maintained, then, unless A has eigenvalues on the order of machine precision or has a large condition number, convergence $x^k \to A^{-1}b$ is guaranteed [26].

For a Lanczos eigensolver in finite precision, when the ith Ritz pair $(\theta_i^k, V_k y^i)$ converges at iteration k, (2.5) shows that $\eta_k |y_k^i| \approx 0$. Unfortunately,

$$v^{kT} V_k y^i \leq \frac{\epsilon ||A||}{\eta_k |y_k^i|} \tag{2.7}$$

is also true [20, 26, 32], where ϵ is machine precision. Thus, in the face of finite precision, the newest Lanczos vector v^k loses orthogonality with the others when some Ritz value has converged to an eigenvalue of A, and the unwanted component of v^k is in the direction of the converged Ritz vector $V_k y^i$. Now the CG–Lanczos relation in (2.6) explains why both CG and the CG sampler experience loss of orthogonality of the residuals and a corresponding loss of conjugacy in the search directions.

Remark 2 and (2.7) show that loss of conjugacy can happen at the same time that CG converges, but it can also happen before (as in one of the examples in section 4). The upside is that, by the time CG converges, some Lanczos eigenpairs have already converged. It is well known which eigenpairs of A are being estimated.

REMARK 3. *The eigenvalues of* A *that are best approximated by the converged Ritz values* $\{\theta_i^k\}$ *are the extreme ones and the well-separated ones* [26, 32, 40, 47].

As iterations continue past convergence of some of the Lanczos Ritz pairs and the corresponding loss of orthogonality, *ghost eigenvalues* of T_k appear, which estimate eigenvalues of A which have already been estimated by earlier Ritz values [14, 20, 50]. That is, T_k has clustered eigenvalues near an isolated eigenvalue of A. Equation (2.7) partially explains this phenomenon, showing that the newer Ritz vectors leak back into the eigenspaces spanned by previous Ritz vectors.

3. Accuracy of the CG covariance approximations. A conjugate direction sampler behaves like a Lanczos eigensolver in finite precision. That is, search directions are doomed to lose conjugacy at some iteration $k < n$. The main theorem of this section makes the connection between Lanczos and the covariance matrices of the CG samples explicit, which describes why, without corrective measures, loss of conjugacy prohibits sampling from the full Gaussians of interest. Nevertheless, the CG sampler produces an approximate sample from $N(0, A)$ with a realized covariance which is the best k-rank approximation to A in the same k-dimensional Krylov space searched by the CG linear solver. The CG sampler also produces an approximate sample from $N(0, A^{-1})$, which has a realized covariance that is the best k-rank approximation to

A^{-1} in the same Krylov space. Like the difficulty faced by iterative eigenproblem solvers, the accuracy of the Krylov k-rank approximations of A and A^{-1} depends on the distribution of the eigenvalues of A.

3.1. Eigenpairs of the CG covariances are Ritz pairs. We saw in section 2.2 that in exact arithmetic, the CG sampler generates samples

$$y^k|(y^0,b) \sim N(y^0, P_k D_k^{-1} P_k^T) \quad \text{and} \quad c^k|(y^0,b) \sim N(Ay^0, AP_k D_k^{-1} P_k^T A),$$

and that these distributions converge to $N(0, A^{-1})$ and $N(0, A)$, respectively, as $k \to n$ when A has distinct eigenvalues. How good are these distribution approximations after $k < n$ iterations when the conjugate directions used by the CG sampler lose conjugacy?

This question has already been answered for $y^k|(y^0,b)$. Lemma 2.1.2 shows that $\mathrm{Var}(y^k|y^0,b)$ is similar to T_k^{-1}, with eigenvalues that are the reciprocals of the Ritz values, $\{1/\theta_i^k\}$. From (2.4) it follows that the eigenvectors of $\mathrm{Var}(y^k|y^0,b)$ are the Ritz vectors $\{V_k q^i\}$, the Lanczos estimates of the eigenvectors of A.

Characterizing $\mathrm{Var}(c^k|b,y^0)$ when the CG sampler terminates with $r^k = 0$ is straightforward as well. In this case, Lemma 2.1.3 and (2.3) show that the CG sampler has converged to an invariant subspace, $AV_k = V_k T_k$, which shows that $\mathrm{Var}(c^k|y^0,b) = V_k T_k V_k^T$. Dealing with the case when $r^k \neq 0$, addressed in the following theorem, requires a little more work. The theorem holds as long as conjugacy of the search directions, and orthogonality of the residuals and Lanczos vectors, is maintained.

THEOREM 3.1. *The covariance matrix of the CG sample y^k is*

$$\mathrm{Var}(y^k|y^0,b) = V_k T_k^{-1} V_k^T,$$

and it has k nonzero eigenvalues which are the Lanczos estimates of the eigenvalues of A^{-1}, $\sigma_i(\mathrm{Var}(y^k|y^0,b)) = 1/\theta_i^k$. The eigenvectors of $\mathrm{Var}(y^k|y^0,b)$ are the Ritz vectors $V_k q^i$ which estimate the eigenvectors of A. The covariance matrix of $c^k = Ay^k$ is

$$\mathrm{Var}(c^k|y^0,b) = V_k T_k V_k^T + \eta_k (v^k v^{(k-1)T} + v^{k-1} v^{kT}) + \left|\frac{\beta_k}{\gamma_{k-1}}\right| v^k v^{kT} \tag{3.1}$$

with $||\mathrm{Var}(c^k|y^0,b)||_2 \leq \theta_k^k + |\beta_k/\gamma_{k-1}|$. When $||r^k||_2 = 0$, then

$$\mathrm{Var}(c^k|y^0,b) = V_k T_k V_k^T,$$

and the k eigenpairs of $\mathrm{Var}(c^k|y^0,b)$ with nonzero eigenvalues are the Lanczos Ritz pairs $(\theta_i^k, V_k q^i)$.

Proof. See Appendix A.2 for the derivation of 3.1. When $||r^k||_2 = 0$, Lemma 2.1.3 shows that $\eta_k = \beta_k = 0$, and consequently the eigenpairs of $\mathrm{Var}(c^k|y^0,b)$ are the Lanczos Ritz pairs. ∎

It is shown in [32] that T_k is a Rayleigh quotient,

$$T_k = \underset{\zeta \in \Re^{k\times k}}{\mathrm{argmin}} ||AV_k - V_k\zeta||_2.$$

That is, $V_k T_k V_k^T$ is the best k-rank approximation of A in $\mathcal{K}^k(A, r^0)$ for any k. Since the CG sampler drives the residual to zero at iteration k, the covariance matrices of the CG samples satisfy this same optimality criterion. The following corollary follows from Theorem 3.1 and Remark 2.

COROLLARY 3.2. *In exact arithmetic, if the CG sampler terminates with* $||r^k||_2 = 0$*, then all Ritz pairs* $\{(\theta_i^k, V_k y^i)\}_{i=1}^k$ *have converged to* k *eigenpairs* $\{(\lambda_j, w^j)\}$ *of* A *and*

$$(A^{-1} - \mathrm{Var}(y^k|y^0, b))v = 0 \quad \text{and} \quad (A - \mathrm{Var}(c^k|y^0, b))v = 0$$

for any $v \in \mathcal{K}^k(A, r^0)$.

REMARK 4. *In light of Remark* 3*, Corollary* 3.2 *shows that when the CG sampler terminates with* $||r^k||_2 = 0$, $\mathrm{Var}(y^k|y^0, b)$ *and* $\mathrm{Var}(c^k|y^0, b)$ *are the best* k*-rank approximations to* A^{-1} *and* A*, respectively, in the eigenspaces corresponding to the extreme and well-separated eigenvalues of* A*. In other words, the CG sampler has successfully sampled from these eigenspaces.*

Another consequence of Theorem 3.1 (which follows from Weyl's theorem [32] and the triangle inequality) is that the 2-norm of the error in covariance estimation is at least as large as $1/\lambda_y$, the largest eigenvalue of A^{-1} not being estimated by the Lanczos estimates $\{1/\theta_i^k\}_{i=1}^k$, and it can get as large as this eigenvalue plus the error in the Lanczos estimates,

$$1/\lambda_y \le ||A^{-1} - \mathrm{Var}(y^k|y^0, b)||_2 \le 1/\lambda_y + ||W_k \Lambda_k^{-1} W_k^T - V_k T_k^{-1} V_k^T||_2. \tag{3.2}$$

In this equation, if $\{(1/\lambda_j, w^j)\}$ are the k eigenpairs of A^{-1} being estimated by the Lanczos pairs $\{(1/\theta_i^k, V_k q^i)\}_{i=1}^k$, then W_k is an $n \times k$ matrix with the orthonormal vectors $\{w^j\}$ as columns, and $\Lambda_k = \mathrm{diag}(\{\lambda_j\})$. In exact arithmetic, when the CG sampler terminates with $||r^k||_2 = 0$, Corollary 3.2 shows that

$$||A^{-1} - \mathrm{Var}(y^k|y^0, b)||_2 = 1/\lambda_y.$$

Similarly, if λ_c is the largest eigenvalue of A not being estimated by the Ritz values $\{\theta_i^k\}_{i=1}^k$, then

$$\begin{aligned} \lambda_c \le ||A - \mathrm{Var}(c^k|y^0, b)||_2 \le \lambda_c + \Big|\Big| & W_{-k} \Lambda_{-k}^{-1} W_{-k}^T - V_k T_k V_k^T \\ & - \eta_k (v^k v^{(k-1)T} + v^{k-1} v^{kT}) - \left|\frac{\beta_k}{\gamma_{k-1}}\right| v^k v^{kT} \Big|\Big|_2. \end{aligned} \tag{3.3}$$

In this equation, if $\{(\lambda_j, w^j)\}$ are the k eigenpairs of A being estimated by the Lanczos Ritz pairs $\{(\theta_i^k, V_k q^i)\}_{i=1}^k$, then W_{-k} is an $n \times k$ matrix with the orthonormal vectors $\{w^j\}$ as columns, and $\Lambda_{-k} = \mathrm{diag}(\{\lambda_j\})$. By Theorem 3.1 and Corollary 3.2, if the CG sampler terminates with $||r^k||_2 = 0$, then $||A - \mathrm{Var}(c^k|y^0, b)||_2 = \lambda_c$.

Equation (3.1) given in Theorem 3.1 also applies to the conjugate direction sampler given by Schneider and Willsky [44, eq. 33]. However, the rest of the theorem does not apply, since that sampler does not drive the CG residual to zero.

3.2. CG sampling in finite precision. In finite precision, we saw in section 2.4 that loss of conjugacy can happen at the same time that CG converges with a small residual, in which case the optimality of the CG sample covariance matrices outlined in Theorem 3.1 and Remark 4 holds. On the other hand, loss of conjugacy can occur before the CG sampler drives the residual to zero. In this case, at the iteration when loss of conjugacy occurs, Theorem 3.1 still assures that $\mathrm{Var}(y^k|y^0, b)$ is the best approximation to A^{-1} in the Krylov subspace which contains the converged Ritz vectors.

If loss of conjugacy occurs at some iteration k before the CG residual has converged to zero, (3.1) describing $\mathrm{Var}(c^k|y^0, b)$ in Theorem 3.1 holds. The magnitudes of

the scalars $|\beta_k/\gamma_{k-1}|$ and η_k in this case indicate how well $\mathrm{Var}(c^k|y^0, b)$ approximates A in the Krylov subspace which contains the converged Ritz vectors.

The Kaniel–Paige–Saad theory suggests that Ritz pairs will converge to the extreme eigenpairs by iteration $k = 2\sqrt{n}$ [32], and so loss of conjugacy of the search directions and loss of orthogonality of the CG residuals will occur by then as well. Thus, without remedy, Lanczos is useful for accurately estimating *at most* $k = 2\sqrt{n}$ eigenpairs of A. By Theorem 3.1, the CG sampler is guaranteed to sample from *at least* these eigenspaces.

The numerical examples in section 4 suggest that after loss of conjugacy, running the CG sampler until the residual is small does not have a deleterious effect on the samples. On the contrary, the CG sampler does continue to sample from new eigenspaces, since $\mathrm{Var}(y^k)$ better approximates A^{-1}, and $\mathrm{Var}(c^k)$ better approximates A as iterations continue to drive the residual to zero.

3.3. $\mathrm{Var}(y^k)$ is the CG polynomial. Viewing the CG sampler as a Lanczos process allows $\mathrm{Var}(y^k|y^0, b)$ to be described with respect to the CG polynomial described in [3, 30, 31]. From the fact that $\mathrm{range}(P_{k+1}) = \mathcal{K}^{k+1}(A, r^0)$, we can re-express the conjugate directions in terms of the Krylov basis:

$$P_{k+1}\gamma^{k+1} = \sum_{i=0}^{k} \tau_i A^i r^0 = \varphi_k(A) r^0,$$

where $\{\tau_i\}$ are scalars, $\gamma^k := [\gamma_0, \ldots, \gamma_{k-1}]^T$ is the vector of one-dimensional minimizers from the CG algorithm, and φ_k is the kth order CG polynomial. Thus, the CG optimizer can be written as

$$x^{k+1} = x^0 + P_{k+1}\gamma^{k+1} = x^0 + \varphi_k(A) r^0.$$

We have the following theorem.

THEOREM 3.3. *The CG polynomial $\varphi_{k-1}(A)$ is equivalent to* $\mathrm{Var}(y^k)$ *in* $\mathcal{K}^k(A, r^0)$. *That is,* $v^T(\mathrm{Var}(y^k) - \varphi_{k-1}(A))v = 0$ *for every* $v \in \mathcal{K}^k(A, r^0)$.

Proof. As in [26, 31], define the kth order *Lanczos polynomial* as

$$\nu_k(\theta) := 1 - \varphi_{k-1}(\theta)\theta,$$

which is shown to be proportional to the characteristic polynomial of T_k in [26, p. 76]. Since $\{(\theta_i^k, q^i)\}$ are the eigenpairs of T_k, $\nu_k(\theta_i^k) = 0$, which implies that $\nu_k(\theta_i^k)q^i = \nu_k(T_k)q^i = 0$, and so the CG polynomial $\varphi_{k-1}(T_k)$ has $\left(1/\theta_i^k, q^i\right)$ as eigenpairs:

$$\varphi_{k-1}(T_k)q^i = T_k^{-1}(I - \nu_k(T_k))q^i = T_k^{-1}q^i = \frac{1}{\theta_i^k}q^i.$$

Since $T_k = V_k^T A V_k$ and V_k has orthonormal columns, by Theorem 3.1 it holds that $v^T(\varphi_{k-1}(A) - \mathrm{Var}(y^k|y^0, b))v = 0$ for any $v \in \mathrm{range}(V_k) = \mathcal{K}^k(A, r^0)$. ∎

Theorem 3.3 suggests that polynomial preconditioners which estimate the characteristic polynomial of A for linear solvers [19, 31, 39, 41, 48] can be used for samplers by damping out the eigenspaces that have already been sampled, much like the potentially expensive reorthogonalization [20, 32] methods used to maintain orthogonality in Lanczos. We investigate this possibility elsewhere.

3.4. Using traces to assess the accuracy of the CG covariances. In (3.2) and (3.3) we bounded the 2-norm of the covariance error for each of the two types of CG samples and showed that when the CG sampler converges in exact arithmetic, the 2-norm is equal to the largest eigenvalue not being estimated by the underlying Lanczos process. We do not know of an inexpensive way to explicitly calculate these bounds for large problems. Instead, we now provide an inexpensive yet potentially coarse Monte Carlo measure of how accurate the covariances of the CG samples are.

In *principal component analysis* (also called a *Karhunen–Loeve decomposition*), a common way to assess the appropriateness of representing n-dimensional data by projections onto $k < n$ eigenvectors of the sample covariance matrix is to consider the ratio of the trace of the covariance of the projections to the trace of the original covariance matrix [35]. The ratio of traces is equal to the fraction of mean-squared error reduction [44]. In the context of the CG sampler, a computationally trivial way to assess how well the covariance matrix A is represented by $\mathrm{Var}(c^k|y^0, b)$ is to compare $\mathrm{trace}(A)$ to $\mathrm{trace}(\mathrm{Var}(c^k|y^0, b))$. Since Ritz values interlace the eigenvalues of $\mathrm{Var}(c^k|y^0, b)$ (see Lemma A.1), $\mathrm{trace}(T_k) \leq \mathrm{trace}(\mathrm{Var}(c^k|y^0, b))$ (with equality if $||r^k||_2 = 0$), which yields the lower bound on the proportion of the variance not described by $\mathrm{Var}(c^k|y^0, b)$,

$$\frac{\sum \alpha_i}{\mathrm{trace}(A)} \leq \frac{\mathrm{trace}(\mathrm{Var}(c^k|y^0, b))}{\mathrm{trace}(A)}. \tag{3.4}$$

This ratio is similar to the stopping criterion for the sampler presented in [44], and should be calculated at the last CG sampler iteration before conjugacy of the search directions is lost. Lemma 2.1.3 shows how $\{\alpha_i\}$ can be computed from the CG sampler.

The proportion of the variance A^{-1} represented by $\mathrm{Var}(y^k|y^0, b)$ is quantified by $\mathrm{trace}(\mathrm{Var}(y^k|y^0, b))/\mathrm{trace}(A^{-1})$. By Theorem 3.1, this is $\mathrm{trace}(T_k^{-1})/\mathrm{trace}(A^{-1})$. The numerator $\mathrm{trace}(T_k^{-1})$ should be calculated at the last CG sampler iteration before conjugacy of the search directions is lost. Computing T_k^{-1} directly is not necessary since

$$\mathrm{trace}(T_k^{-1}) = \sum_{j=0}^{k-1} \frac{1}{||r^j||_2^2} \sum_{i=j}^{k-1} \gamma_i ||r^i||_2^2 \tag{3.5}$$

(see Appendix A.3), and so is easily calculated by maintaining a cumulative sum from quantities available from the CG sampler.

Since A is large, $\mathrm{trace}(A^{-1})$ cannot be calculated directly. Fortuitously, a Lanczos procedure is well suited to this task [5, 26]. When initialized with $v^0 = u/||u||_2$, Lanczos implements *Gaussian quadrature* to estimate $u^T f(A)u$ for any vector u and any function f of A:

$$u^T f(A)u \approx ||u||_2^2 \sum_{i=1}^{k} (q_1^i)^2 f(\theta_i^k) = ||u||_2^2 e^{1T} f(T_k) e^1.$$

That is, the nodes for Gaussian quadrature are the Ritz values $\{\theta_i^k\}$, and the weights are the squares of the first components of the eigenvectors of T_k (times $||u||_2^2$). More specifically, when initialized with $b = u$ and $y^0 = 0$, CG and the CG sampler implement Gaussian quadrature to estimate

$$u^T A^{-1} u \approx ||r^0||_2^2 [T_k^{-1}]_{1,1} = \sum_{i=0}^{k-1} \gamma_i ||r^i||_2^2. \tag{3.6}$$

The equality in (3.6) follows from Lemma 2.1.2 (see Appendix A.3).

An infeasible way to use (3.6) to approximate trace(A^{-1}) is to run CG with $u = e^j$ to estimate $[A^{-1}]_{jj} = e^{jT}A^{-1}e^j$ for each j. To estimate trace(A^{-1}) in one Lanczos run, an ingenious Monte Carlo approach is given in [5], for which Lanczos needs to be initialized with an $n \times 1$ random vector u of -1's and 1's (with equal probability of each, denoted hereafter as $u = (\pm 1, \pm 1, \dots)$). In fact, initializing Lanczos with any random vector composed of independent, zero mean, and unit variance entries yields an unbiased trace estimate, but $u = (\pm 1, \pm 1, \dots)$ has minimum variance [49]. When the CG sampler is run m times to generate m independent samples, the individual trace estimators from each run can be averaged together to yield an estimator with variance decreased by a factor of $1/m$. The number of runs m required to reach a specified level of precision depends on the spread of the eigenvalues of A [49].

In the examples in section 4, we initialize the CG sampler with either a random vector $b = u = (\pm 1, \pm 1, \dots)$ or $b \sim N(0, I)$, use (3.5) (before loss of conjugacy) to compute trace(T_k^{-1}), and then use (3.6) to estimate trace(A^{-1}). In this way, the CG sampler can estimate the proportion of the variance in A^{-1} explained by Var(y^k) at an additional expense of a vector inner product per iteration. Calculating the proportion of the variance in A explained by Var(c^k) using (3.4) is trivial.

3.5. Intrinsic Gaussian distributions. It is sometimes desirable to draw samples from Gaussians with singular precision or covariance matrices. Examples of such distributions are the *intrinsic* Gaussian distributions resulting from a locally linear definition of the precision matrix of a Gaussian Markov random field (GMRF) in a region with a boundary [22, 38], or as conditional distributions [10]. These distributions are improper, having a precision matrix that is symmetric positive semidefinite with a nontrivial nullspace null(A), making the density invariant under addition of any element of null(A). Hence the mean and covariance are undefined (or unbounded). In typical applications the component of the nullspace in a sample is fixed, usually at 0, in which case the mean and covariance are defined with the covariance matrix being the (Moore–Penrose) generalized inverse of the precision matrix.

The CG sampler in Algorithm 1 will draw samples from an intrinsic Gaussian, without modification. Consider the case when the precision matrix has nontrivial nullspace null(A). The residual at each step r^k is orthogonal to null(A), i.e., $r^k \perp$ null(A) since null(A) = null(A^T), and hence by induction all of the conjugate directions are also orthogonal to the nullspace, $p^k \perp$ null(A). Simply initializing $y^0 = x^0 = 0$ gives $x^0 \perp$ null(A), and by induction the CG minimizer x^k satisfies $x^k \perp$ null(A) for every k. Similarly, the CG sample $y^k \perp$ null(A) for all k. Thus, in exact arithmetic, when the algorithm terminates with $k = n - \dim(\text{null}(A))$, the resulting sample is correctly distributed with a zero component in null(A), as shown in section 2.1. More generally, initializing the CG sampler with y^0 which has a component $y_{\text{null}} \in$ null(A) gives a sample y^k with component y_{null} in null(A). Effectively, the algorithm does not "see" null(A) and operates entirely in the orthogonal complement of null(A).

In finite precision, zero eigenvalues of A may effectively be small nonzero values, and vice versa. Hence, for matrices with a large condition number, the notion of a Moore–Penrose generalized inverse is not robust to numerical error, and the CG sampler may not produce samples distributed according to the desired intrinsic Gaussian distribution. Note that the convergence theory we have given for the CG sampler holds only for symmetric positive definite matrices.

4. Examples. We next consider two illustrative examples, for which we used the CG sampler to generate samples $y^k \dot{\sim} N(0, A^{-1})$ and $c^k \dot{\sim} N(0, A)$ to numerically investigate the difference between the theoretical result in exact arithmetic that the

CG sampler produces samples from $N(0, A)$ and $N(0, A^{-1})$, and the realized distributions of the CG samples in finite precision. As we have seen, the reason there is a difference is the loss of conjugacy of the sampling directions, and the accuracy of the realized covariance of the CG samples depends on the distribution of the eigenvalues of the covariance matrix.

The first example we consider is a covariance model $A = \Sigma$, commonly used to account for dependencies among measurements close in space or time, that exhibits the characteristic sharp drop in covariance at some critical point. The eigenvalue distribution for this type of covariance matrix is ideal for the CG sampler: the large eigenvalues of A are well separated, while the small eigenvalues are clustered together near zero. We chose a one-dimensional domain to easily visualize the samples, and a small 100×100 example was used so that convergence of the CG covariance matrices with respect to the 2-norm could be easily monitored. In the second example, we considered a common sparse precision matrix model $A = \Sigma^{-1}$ which presumes that points in space or time are conditionally independent if they are in different neighborhoods. For computational efficiency, we used a small 100×100 precision matrix. For a 10^4 example with this same precision model, the qualitative differences between a Cholesky sample and a CG sample are shown, the latter with a realized covariance which captures only 80% of the variability in the full space.

In addition to the matrix A, the CG sampler requires two initial vectors, y^0 and b. In all of the simulations presented, $y^0 = 0$. Two different values for b were considered. In the first case, each component of b was randomly assigned -1 or 1, each with probability .5; then this $b = (\pm 1, \pm 1 \dots)$ was fixed and used in 10^5 sampler runs. This initialization with a fixed vector b allowed us to numerically assess the accuracy of the CG covariances $\mathrm{Var}(y^k | y^0 = 0, b)$ (see (2.1)) and $\mathrm{Var}(c^k | y^0 = 0, b)$. In our experience, the numerical results using other fixed vectors b are similar to those presented here. The second initialization considered was for $b \sim N(0, I)$, where b was randomly chosen for each of the 10^5 sampler runs. This initialization was chosen in order to numerically assess the effect of using a random vector $b \sim N(0, I)$ on the CG covariances, $\mathrm{Var}(y^k) = \mathrm{E_b}(\mathrm{Var}(y^k | y^0 = 0, b))$ (see (2.2)) and $\mathrm{Var}(c^k) = \mathrm{E_b}(\mathrm{Var}(c^k | y^0 = 0, b))$. Each of the two initializations allows the CG sampler to estimate $\mathrm{trace}(\mathrm{Var}(y^k))/\mathrm{trace}(A^{-1})$, the proportion of variability in A^{-1} described by the CG covariance, in each run (see section 3.4).

4.1. Squared exponential covariance function. The example in this section shows how the CG sampler can be used to generate a sample from a Gaussian process with a specified covariance $A = \Sigma$. Gaussian processes with a squared exponential covariance (or Gaussian covariance) model over some domain $\mathcal{S}$ are commonly used in applications [22, 25, 29, 34, 45, 52, 55]. Consider the Gaussian covariance matrix Σ with elements

$$\Sigma_{ij} = 2 \exp\left(-\frac{(s_i - s_j)^2}{2(1.5)^2} \right) + \epsilon \delta_{ij}$$

for 100 regularly spaced values s_i in the one-dimensional domain $\mathcal{S} = [-3, 3]$, with $\epsilon = 10^{-6}$, and $\delta_{ij} = 1$ if $i = j$, and zero otherwise. Setting $A := \Sigma$, we calculated 10^5 samples from each of the 100-dimensional Gaussians $N(0, A)$ and $N(0, A^{-1})$ using both the CG sampler and a Cholesky factorization. The results are given in Figure 4.1 and in Tables 4.1 and 4.2. Note that A is 100×100, $||A||_2 = 103.5$, $||A^{-1}||_2 = 10^6$, and the condition number is $\kappa(A) = ||A||_2 ||A^{-1}||_2 \approx 10^8$.

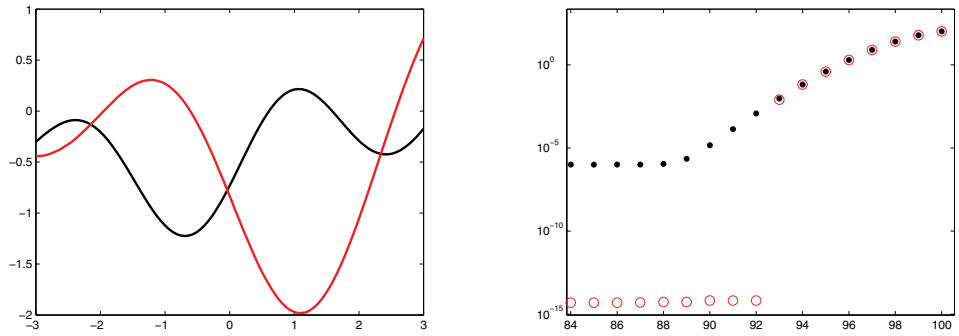

FIG. 4.1. *In the left panel, a Cholesky sample is compared with a CG sample from* $N(0, A)$ *(after* $k = 60$ *CG sampler iterations) for the* 100×100 *Gaussian covariance matrix* A*. In the right panel, the eigenvalues of the* 100×100 *Gaussian covariance* A *are shown as dots, in ascending order, versus the indices* 84 *to* 100*. The open circles are the eigenvalues of the realized CG sample variance* $\mathrm{Var}(c^k)$ *at iteration* $k = 8$*, right before loss of conjugacy. For ease of viewing, the* 92 *zero eigenvalues of* $\mathrm{Var}(c^8)$ *have been depicted near* 10^{-15}*.*

For $0 < \epsilon \leq 10^{-16}$, A is ill-conditioned, and a Cholesky factorization (and the conventional sampling approach) is not possible using MATLAB 2006a. When the diagonal additive component is $\epsilon = 0$, A has a large dimensional nullspace, and Cholesky factors do not even exist. As suggested by the discussion in section 3.5 for positive semidefinite matrices, the CG sampler does produce samples with the desired smoothness for $0 \leq \epsilon \leq 10^{-16}$, although with large amplitudes on the order of 10^5 (not shown), possibly due to contributions from the nullspace null(A).

Over the 10^5 samples with the initialization $b = (\pm 1, \pm 1, \dots)$, the CG sampler ran an average of $k = 60$ iterations (see row 2 of Table 4.1) and terminated when the CG residual was small, $||r^k||_2 < 10^{-4}$. Due to finite precision, conjugacy of the sampling directions was lost at iteration $k = 9$. We calculated an additional 10^5 samples by forcibly terminating the CG sampler at iteration $k = 8$ (see row 1 of Table 4.1). The results show that the realized variance of the CG sample $c^k | \, (b = (\pm 1, \pm 1, \dots))$, estimated from the 10^5 samples, is very close to A, regardless of whether the sampler was terminated at iteration $k = 8$ or $k = 60$. Although this level of accuracy at iteration $k = 8$ is not necessarily assured since $||r^8||_2 = 57$, Theorem 3.1 still predicts that $A \approx \mathrm{Var}(c^8)$ since $|\beta_8/\gamma_7| = .0093$ and $\eta_8 = .2756$.

For the 10^5 runs using the second initialization with $b \sim N(0, I)$, it was clear that the CG sampler must be run past loss of conjugacy to get $r^k \approx 0$ (see the third and fourth rows of Table 4.1).

Remark 4 shows that the realized CG covariance matrices are accurate only in the eigenspaces corresponding to the well-separated eigenvalues of A. Figure 4.1 shows that the large eigenvalues of A are well separated with many eigenvalues near $\epsilon = 10^{-6}$. This explains why $\mathrm{Var}(c^k)$ approximates A well. By the 8th CG sampler iteration, the eigenspaces corresponding to the largest 8 eigenvalues have already been sampled, and so already, $\mathrm{Var}(b^8)$ approximates A, accounting for over 99% of the variability in the 100-dimensional $N(0, A)$. The fact that loss of conjugacy coincides with accurate approximations of these eigenpairs of A is a hallmark of Lanczos methods (section 2.4). After loss of conjugacy, subsequent conjugate directions leak back into these 8 eigenspaces, which illustrates the failure of the CG sampler, and of Krylov methods

TABLE 4.1
Comparing sample covariance matrices $\mathrm{Var}(c^k)$ *of Cholesky and CG samples* $c^k \dot{\sim} N(0, A)$ *when* A *is a squared exponential (Gaussian) covariance matrix, with* $||A||_2 = 103.5$. *Each row in the table summarizes the results from* 10^5 *samples.*

Method	b	k	$\|\|\mathrm{Var}(c^k)\|\|_2$	$\frac{\|\|A-\mathrm{Var}(c^k)\|\|_2}{\|\|A\|\|_2}$	$\frac{\mathrm{trace}(\mathrm{Var}(c^k))}{\mathrm{trace}(A)}$
CG sampler	$(\pm 1, \pm 1, \dots)$	8	102.7	.0081	.9965
		60	104.0	.0059	1
	$N(0, I)$	8	62895	606.7	–
		57	103.7	.0058	.9980
Cholesky	–	–	103.3	.0044	1

TABLE 4.2
Comparing sample covariance matrices $\mathrm{Var}(y^k)$ *of Cholesky and CG samples* $y^k \dot{\sim} N(0, A^{-1})$ *when* A *is a squared exponential (Gaussian) precision matrix,* $||A^{-1}||_2 = 10^6$. *Each row in the table summarizes the results from* 10^5 *samples.*

Method	b	k	$\|\|\mathrm{Var}(y^k)\|\|_2$	$\frac{\|\|A^{-1}-\mathrm{Var}(y^k)\|\|_2}{\|\|A^{-1}\|\|_2}$	$\frac{\mathrm{trace}(\mathrm{Var}(y^k))}{\mathrm{trace}(A^{-1})}$
CG sampler	$(\pm 1, \pm 1, \dots)$	8	7348	1	.0001
		60	999,580	1	.0291
	$N(0, I)$	8	706.9	.9995	.0006
		57	795,809	.9895	.0285
Cholesky	–	–	1,061,648	.0621	1

in general, to generate search directions which span all n eigenspaces of A. However, Tables 4.1 and 4.2 show that the sampling directions generated past loss of conjugacy do span some new eigenspace(s), since the CG samples y^k and c^k for $k > 8$ have realized covariances that are closer to A^{-1} and A, respectively.

Theorem 3.1 also predicts that $\mathrm{Var}(y^k)$ is a poor representation of A^{-1} since 10^{-6} is a repeated eigenvalue of A. Even after preconditioning as in [17], these repeats remain clustered relative to the large eigenvalues of A. Table 4.2 confirms that the realized variance of the CG sample y^k is a poor approximation to A^{-1}, with $||A^{-1} - \mathrm{Var}(y^k)||_2 \approx ||A^{-1}||_2 = 10^6$. Although y^k is not of concern when the Gaussian of interest is $N(0, A = \Sigma)$, this result indicates the inability of the CG sampler to generate samples using a precision matrix model which has many small eigenvalues clustered together.

4.2. Second order locally linear precision matrix. The CG sampler can also be seeded with a precision matrix $A = \Sigma^{-1}$. We next consider a GMRF with a specified precision matrix, a common approach in spatial applications, since the underlying Markov assumptions yield a sparse precision matrix. The following precision matrix Σ^{-1} is considered in [22, 38]:

$$[\Sigma^{-1}]_{ij} = 10^{-3}\delta_{ij} + \begin{cases} n_i & \text{if } \ i = j, \\ -1 & \text{if } \ i \neq j \ \text{ and } \ ||s^i - s^j||_2 < 1.5, \\ 0 & \text{otherwise.} \end{cases}$$

The discrete points s^i are on a regular 10×10 unit grid over the two-dimensional domain $\mathcal{S} = [1, 10] \times [1, 10]$. The number of points in a neighborhood of radius 1.5 centered at s^i, but not including s^i, is given by n_i (e.g., $n_1 = 3, n_2 = 5, n_{12} = 8$). The CG sampler was initialized with the 100×100 matrix $A := \Sigma^{-1}$ and $y^0 = 0$ and ran 10^5 times with $b = (\pm 1, \pm 1, \dots)$ and another 10^5 times with $b \sim N(0, I)$. The CG sampler ran an average of $k = 37$ iterations under both initial conditions and

TABLE 4.3

Comparing sample covariance matrices $\mathrm{Var}(y^k)$ *of Cholesky and CG samples* $y^k \dot{\sim} N(0, A^{-1})$ *when* A *is a second order locally linear precision matrix with* $||A^{-1}||_2 = 10^3$ *and* $\mathrm{trace}(A^{-1}) = 1.16 \times 10^4$. *Each row in the table summarizes the results from* 10^5 *samples, except for the second entry for Gibbs (see text).*

Method	b	k	$\|\|\mathrm{Var}(y^k)\|\|_2$	$\frac{\|\|A^{-1}-\mathrm{Var}(y^k)\|\|_2}{\|\|A^{-1}\|\|_2}$	$\frac{\mathrm{trace}(\mathrm{Var}(y^k))}{\mathrm{trace}(A^{-1})}$
CG sampler	$(\pm 1, \pm 1, \dots)$	37	1003.93	.0040	.9902
	$N(0, I)$	37	1006.17	.0062	.9923
PCG sampler	$(\pm 1, \pm 1, \dots)$	43	985.27	.0148	.9722
		61	997.86	.0022	.9885
	$N(0, I)$	43	998.30	.0082	.9789
		61	1009.52	.0095	.9997
Gibbs	$(\pm 1, \pm 1, \dots)$	223	957.62	.7762	–
		1.49×10^5	–	.0040	–
Cholesky	–	–	1001.48	.0017	1

TABLE 4.4

Comparing sample covariance matrices $\mathrm{Var}(c^k)$ *of Cholesky and CG samples* $c^k \dot{\sim} N(0, A)$ *when* A *is a second order locally linear covariance matrix with* $||A||_2 = 11.61$ *and* $\mathrm{trace}(A) = 684.1$. *Each row in the table summarizes the results from* 10^5 *samples.*

Method	b	k	$\|\|\mathrm{Var}(c^k)\|\|_2$	$\frac{\|\|A-\mathrm{Var}(c^k)\|\|_2}{\|\|A\|\|_2}$	$\frac{\mathrm{trace}(\mathrm{Var}(c^k))}{\mathrm{trace}(A)}$
CG sampler	$(\pm 1, \pm 1, \dots)$	37	11.66	1	.3187
	$N(0, I)$	37	11.30	.6975	.3182
PCG sampler	$(\pm 1, \pm 1, \dots)$	43	11.14	.9311	.4425
		61	11.48	.8794	.5776
	$N(0, I)$	43	11.13	.7478	.4713
		61	11.46	.6606	.5764
Cholesky	–	–	11.64	.0428	1

terminated when $||r^k||_2 < 10^{-4}$. Conjugacy was never lost. Note that $||A||_2 = 11.61$, $||A^{-1}||_2 = 10^3$, and the condition number is $\kappa(A) = 1.16 \times 10^4$.

Results are given in Tables 4.3 and 4.4. Since the smallest eigenvalues of A are spread out (not shown), the covariance of the CG sample y^k is an accurate approximation of A^{-1}. The rest of the eigenvalues larger than 1 are grouped close together, so the covariance of the CG sample c^k accounts for only about half of the variance in A. Even in this case, $\mathrm{Var}(c^k)$ and A are qualitatively very similar (not shown). Although c^k is not of concern when the GMRF of interest is $N(0, A = \Sigma^{-1})$, this result indicates the inability of the CG sampler to generate samples using a covariance matrix model which has many large eigenvalues clustered together.

At iteration $k = 37$, $|\beta_k/\gamma_{k-1}| = .4045$, so again by Theorem 3.1, the eigenvalues of $\mathrm{Var}(c^k|y^0, b)$ are essentially the Ritz values at loss of conjugacy. Now, however, $\eta_k = 1.9489$, and so the eigenvectors of $\mathrm{Var}(c^k|y^0, b)$ have a larger error.

A PCG sampler was also run with the bidiagonal preconditioner described in [17]. The CG sampler ran an average of $k = 43$ iterations under both initialization conditions before losing conjugacy (on average, $||r^{43}||_2 = .1980$) and terminated when the norm of the CG residual was less than 10^{-4}. Tables 4.3 and 4.4 indicate that the PCG covariance approximations are only slightly better than the CG covariance approximations.

A Gibbs sampler takes as input a precision matrix $A = \Sigma^{-1}$ and generates samples from $N(0, A^{-1})$ at the same cost of $2n^2$ flops per iteration as the CG sampler. We implemented a Gibbs sampler, as described in [9], in the current example. Table 4.3

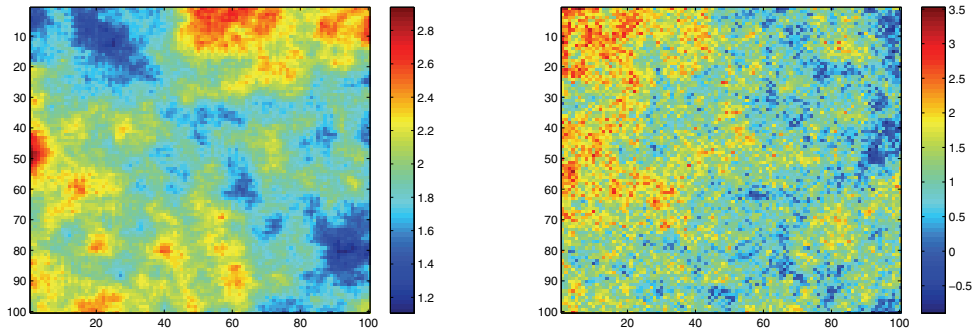

FIG. 4.2. *In the left panel is a CG sample* $y^k \dot{\sim} N(0, A^{-1})$ *from a* 10^4*-dimensional Gaussian over a two-dimensional domain with a second order locally linear precision matrix. The realized variance* $\mathrm{Var}(y^k)$ *accounts for* 80% *of the variability in* A^{-1}*. A Cholesky sample is shown in the right panel.*

shows the results. Although Gibbs samples are guaranteed to converge in distribution to the target $N(0, A^{-1})$, the geometric asymptotic convergence rate is very slow. After $k = 223$ iterations, the diagnostic measures indicate that the realized covariance of the Gibbs samples is far from convergence. A theoretical calculation (based on the spectral radius of the Gibbs stationary linear operator, which is the same as that employed by Gauss–Seidel [9]) was used to ascertain that 1.49×10^5 iterations are required before the relative error of the realized precision matrix will be comparable with that of the CG sampler, $||A^{-1} - \mathrm{Var}(y^k)||_2/||A^{-1}||_2 \approx .0040$.

Figure 4.2 shows samples from a 10^4-dimensional Gaussian with a second order locally linear precision matrix. The CG sampler exited after $k = 315$ iterations when $||r^k|| < 10^{-4}$, and conjugacy of the directions was maintained. The $10^4 \times 10^4$ matrix A has $||A||_2 = 12$, $\kappa(A) = 1.2 \times 10^5$, $\mathrm{trace}(A) = 7.88$, and $\mathrm{trace}(A^{-1}) = 1.38 \times 10^4$. The CG sample y^k is an approximate sample from $N(0, A^{-1})$ since $\mathrm{trace}(\mathrm{Var}(y^k))/\mathrm{trace}(A^{-1}) \approx 0.80$, a quantity which the CG sampler can estimate (see section 3.4). Notice that the CG sample is much smoother than the Cholesky sample given in Figure 4.2, showing how the exclusion of the small eigenvalues of A^{-1} (which account for the other 20% of the variability) affects the CG samples y^k. The variance of the CG sample c^k in this case accounts for only 2.4% of the variability described in A.

5. Conclusions. We have shown how the CG samples $y^k \dot{\sim} N(0, A^{-1})$ and $c^k \dot{\sim} N(0, A)$ can be easily generated by adding a single inexpensive line of code to CG. We have also given an analysis of the CG sampler which describes how it works in practice without any corrective measures. The eigenpairs of $\mathrm{Var}(y^k)$ are the eigenpair estimates of A^{-1} found by Lanczos, and hence $\mathrm{Var}(y^k)$ is a satisfactory k-rank approximation to A^{-1} when the small eigenvalues of A are well separated and the rest of the spectrum is relatively large, as is the case for common precision matrix models (e.g., the locally linear Laplacian [38]). Similarly, the CG sample c^k has a realized covariance matrix which is a satisfactory k-rank approximation to A when the large eigenvalues of A are well separated and the rest of the spectrum is relatively small, which is the case for common covariance models (e.g., exponential and Gaussian covariances over space and/or time [52]). Even when implemented with a matrix A without such eigenvalue distributions, the CG sampler has been effective at generating

samples and obtaining low rank covariance and precision matrix approximations when implemented within the Kalman filter [7, 8].

The advantages of the CG sampler are that one can use the covariance or precision model of choice (i.e., restrictive low rank models [13] and simplifying Markov assumptions [38] are not necessary), the potentially large matrix need never be constructed or factored, and simple numerical checks (presented in this paper) exist to check the accuracy of the k-rank variances of the CG samples produced.

The disadvantage of the CG sampler is that the accuracy of the k-rank realized covariance matrices of the CG samples depends on the eigenvalue distribution of A, which is unknown for large complex models. In order to generate samples with realized covariances which are arbitrarily close to the desired covariance, corrective measures such as reorthogonalization have successfully been used [44]. Depending on the distribution of the eigenvalues of the covariance matrix being modeled, the cost of *full* reorthogonalization can be as much as $\mathcal{O}(n^3)$ flops and can require storage of large dense $n \times \mathcal{O}(n)$ projection matrices. Computationally efficient *semiorthogonalization* implementations are available (e.g., selective and periodic reorthogonalization [4, 14, 20, 32, 46]), but memory requirements still increase linearly with the number of eigenpairs of A that one needs to (implicitly) estimate. Schneider and Willsky report that "the quality of simulation is not significantly affected by the type of orthogonalization used" [44, p. 300]. For large problems when reorthogonalization is not feasible, one could instead try preconditioning to spread out the eigenvalues of A, which is inexpensive when the $n \times n$ preconditioning matrix is sparse. However, as we have seen in the numerical examples, this improves accuracy only marginally. Another option is to use an iterative Gibbs sampler, accelerated by initialization with a CG sample, at a cost of $2n^2$ flops per iteration. Again, however, the caveat is that the geometric convergence of the Gibbs sampler is reduced by only a constant factor, and only when $(A^{-1} - \mathrm{Var}(y^k))$ lies in the eigenspaces corresponding to the small eigenvalues of the Gibbs stationary operator. This is an area of active research.

The disadvantages just outlined, common to any Lanczos process, are due to a failure to maintain an orthogonal basis for the associated Krylov space. Krylov samplers, however, need not suffer from this same issue, since explicit estimates of the eigenpairs are not necessary in order to sample from the associated eigenspaces. For the CG sampler, the equivalence between the realized covariances and the CG and Lanczos polynomials are the impetus for another area of active research, where common inexpensive polynomial preconditioning techniques for linear solvers are applied to generate exact samples from high dimensional Gaussians of interest.

Appendix.

A.1. Proof of Lemma 2.1. To prove the first part of the lemma, define the $k \times k$ diagonal matrices

$$\Delta_k := \mathrm{diag}(||r^0||, \dots, ||r^{k-1}||), \qquad N_k := \mathrm{diag}(1, -1, 1, \dots)$$

and define the upper bidiagonal matrix

$$B_k = \begin{pmatrix} 1 & \beta_1 & & & \\ & \ddots & \ddots & & \\ & & & 1 & \beta_{k-1} \\ & & & & 1 \end{pmatrix}.$$

Now the relation $r^k = p^k + \beta_k p^{k-1}$ can be written as $V_k = P_k B_k \Delta_k^{-1} N_k$, which shows that range$(P_k)$ = range(V_k). Since $r^{kT} p^i = 0$ for every $i < k$, multiplying $p^k = r^k - \beta_k p^{k-1}$ by r^{kT} shows that

$$r^{kT} p^k = r^{kT} r^k. \tag{A.1}$$

Thus, letting $R_k = \Delta_k B_k^{-1} N_k$ yields a QR factorization $P_k = V_k R_k$ since $R_k = V_k^T P_k$ is upper triangular with elements

$$[R_k]_{ij} = (-1)^{i-1} \frac{r^{(i-1)T} p^{(j-1)}}{||r^{i-1}||} \tag{A.2}$$

which are zero for all $i > j$. The second part of the lemma is proved by rewriting the statement of conjugacy $P_k^T A P_k = D_k$ as $R_k^T V_k^T A V_k R_k = R_k^T T_k R_k = D_k$ and so $P_k D_k^{-1} P_k^T = V_k R_k D_k^{-1} R_k^T V_k^T = V_k T_k^{-1} V_k^T$. To prove part 3, define the lower bidiagonal matrix $L_k := R_k^{-T}$ so that $T_k = L_k D_k L_k^T$ (similar to the "LDL^T" factorizations of T_k considered in [6, 14, 20, 26, 32, 33]). Now multiply the right-hand side out and equate the components of the right-hand side with T_k. Part 4 follows from applying (A.1) and (A.2) to the components of $T_k = L_k D_k L_k^T$.

A.2. Proof of Theorem 3.1. Let $\sigma_i(B)$ denote the ith eigenvalue of an $m \times m$ symmetric matrix B, $\sigma_1(B) \le \cdots \le \sigma_m(B)$, and let $\sigma_{-i}(B)$ denote the eigenvalues in descending order $\sigma_{-1}(B) \ge \cdots \ge \sigma_{-m}(B)$.

LEMMA A.1. *The k nonzero eigenvalues of* $\mathrm{Var}_b := \mathrm{Var}(c^k | y^0, b)$ *interlace the Lanczos estimates of the eigenvalues of A,*

$$\theta_{-k}^k \le \sigma_{-k}(\mathrm{Var}_b) \le \theta_{-(k-1)}^k \le \cdots \le \sigma_{-2}(\mathrm{Var}_b) \le \theta_{-1}^k \le \sigma_{-1}(\mathrm{Var}_b) \le \theta_k^k + \left| \frac{\beta_k}{\gamma_{k-1}} \right|.$$

Proof. Consider the decomposition $V_n = [V_k \;\; \widetilde{V}_k]$. Then

$$T_n = V_n^T A V_n = \begin{pmatrix} V_k^T A V_k & V_k^T A \widetilde{V}_k \\ \widetilde{V}_k^T A V_k & \widetilde{V}_k^T A \widetilde{V}_k \end{pmatrix} = \begin{pmatrix} T_k & S^T \\ S & \widetilde{T}_k \end{pmatrix}, \tag{A.3}$$

where $\widetilde{T}_k$ is tridiagonal and $S = \eta_k e^1 e^{kT}$ is an $(n-k) \times k$ matrix, where the only nonzero element is the upper right entry η_k. Now pre- and postmultiplying the equation $P_k D_k^{-1} P_k^T = V_k T_k^{-1} V_k^T$ by A shows that $\mathrm{Var}(c^k | y^0, b)$ has the same eigenvalues as

$$\begin{aligned} T_k^{-1} V_k^T A^2 V_k &= T_k^{-1} V_k^T A V_n V_n^T A V_k = T_k^{-1} [V_k^T A V_k \;\; V_k^T A \widetilde{V}_k] \begin{pmatrix} V_k^T A V_k \\ \widetilde{V}_k^T A V_k \end{pmatrix} \\ &= T_k^{-1} (T_k^2 + S^T S) = T_k + \eta_k^2 T_k^{-1} e^k e^{kT}. \end{aligned}$$

The second term in the last equation is a rank one update. Thus, for $i \le k-1$, we have that [20, 32] $\sigma_i(A P_k D_k^{-1} P_k^T A) \in [\sigma_i(T_k), \sigma_{i+1}(T_k)] = [\theta_i^k, \theta_{i+1}^k]$. Substituting in $T_k^{-1} = Q_k \Theta^{-1} Q_k^T$ (see (2.4)) shows that $\eta_k^2 T_k^{-1} e^k e^{kT}$ has the nonzero eigenvalue

$$\eta_k^2 e^{kT} T_k^{-1} e^k = \eta_k^2 \sum_{i=1}^{k} \frac{(q_k^i)^2}{\theta_i^k} = \left| \frac{\beta_k}{\gamma_{k-1}} \right| \tag{A.4}$$

by Lemma 2.1, parts 3 and 4. Now apply Weyl's monotonicity theorem [32]. ∎

To get the eigenvectors, start with (2.1) and apply (2.3):

$$\begin{aligned}\mathrm{Var}(c^k|y^0,b) &= AP_kD_k^{-1}P_k^TA = AV_kT_k^{-1}V_k^TA \\ &= (V_kT_k+\eta_kv^ke^{kT})T_k^{-1}(V_kT_k+\eta_kv^ke^{kT})^T \\ &= V_kT_kV_k^T+\eta_k(v^kv^{(k-1)T}+v^{k-1}v^{kT})+\left|\frac{\beta_k}{\gamma_{k-1}}\right|v^kv^{kT}, \end{aligned} \tag{A.5}$$

where the last equation follows from (A.4).

A.3. Estimating trace(A^{-1}). The results in section 3.4 follow from the following lemma.

LEMMA A.2. $[T_k^{-1}]_{j+1,j+1} = \frac{1}{||r^j||_2^2}\sum_{i=j}^{k-1}\gamma_i||r^i||^2$.

To see this, (A.1), conjugacy of the $p^0,\dots,p^k$, and the fact that $p^{kT}r^j=0$ when $j>k$ show that for any $0\le j\le k-1$,

$$||r^k||_2^2 = p^{kT}r^k = p^{kT}(b-Ax_{CG}^k) = p^{kT}\left(b-A\left(x^j+\sum_{i=j}^{k-1}\gamma_ip^i\right)\right) = p^{kT}r^j.$$

Now Lemma 2.1.2 shows that $[T_k^{-1}]_{j+1,j+1} = e^{(j+1)T}T_k^{-1}e^{j+1} = e^{(j+1)T}R_kD_k^{-1}R_k^Te^{j+1}$ and $e^{(j+1)T}R_k = e^{(j+1)T}V_k^TP_k = \frac{(-1)^j}{||r^j||}(0,\dots,0,||r^j||^2,\dots,||r^{k-1}||^2)$. The lemma now follows from the definition of γ_i given in Algorithm 1. ∎

REFERENCES

[1] S. L. ADLER, *Over-relaxation method for the Monte Carlo evaluation of the partition function for multiquadratic actions*, Phys. Rev. D, 23 (1981), pp. 2901–2904.

[2] Y. AMIT AND U. GRENANDER, *Comparing sweep strategies for stochastic relaxation*, J. Multivariate Anal., 37 (1991), pp. 197–222.

[3] O. AXELSSON, *Iterative Solution Methods*, Cambridge University Press, Cambridge, UK, 1996.

[4] Z. BAI, J. DEMMEL, J. DONGARRA, A. RUHE, AND H. VAN DER VORST, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, 2000.

[5] Z. BAI, M. FAHEY, AND G. H. GOLUB, *Some large-scale matrix computation problems*, J. Comput. Appl. Math., 74 (1996), pp. 71–89.

[6] Z. BAI AND R. W. FREUND, *A symmetric band Lanczos process based on coupled recurrences and some applications*, SIAM J. Sci. Comput., 23 (2001), pp. 542–562.

[7] J. M. BARDSLEY, A. PARKER, A. SOLONEN, AND M. HOWARD, *Krylov space approximate Kalman filtering*, Numer. Linear Algebra Appl., to appear.

[8] J. BARDSLEY, A. SOLONENY, A. PARKER, H. HAARIO, AND M. HOWARD, *An ensemble Kalman filter using the conjugate gradient sampler*, Int. J. Uncertainty Quantification, to appear.

[9] P. BARONE AND A. FRIGESSI, *Improving stochastic relaxation for Gaussian random fields*, Probab. Engrg. Inform. Sci., 23 (1990), pp. 2901–2904.

[10] J. BESAG AND C. KOOPERBERG, *On conditional and intrinsic autoregressions*, Biometrika, 82 (1995), pp. 733–746.

[11] J. BRANDTS AND H. VAN DER VORST, *The convergence of Krylov methods and Ritz values*, in Conjugate Gradient Algorithms and Finite Element Methods, M. Krizek, P. Neittaanmaki, R. Glowinski, and S. Korotov, eds., Springer, New York, 2004, pp. 47–68.

[12] G. CASELLA AND R. L. BERGER, *Statistical Inference*, 2nd ed., Duxbury, Belmont, CA, 2002.

[13] N. CRESSIE, *Fixed Rank Kriging for Very Large Spatial Data Sets*, Technical report 780, Department of Statistics, The Ohio State University, Columbus, OH, 2006.

[14] J. K. CULLUM AND R. A. WILLOUGHBY, *Lanczos Algorithms for Large Symmetric Eigenvalue Computations: Vol.* I: *Theory*, SIAM, Philadelphia, 2002.

[15] M. DUFLO, *Random Iterative Models*, Springer-Verlag, Berlin, 1997.

[16] T. ERICSSON AND A. RUHE, *The spectral transformation Lanczos method for the numerical solution of large sparse generalized symmetric eigenvalue problems*, Math. Comp., 35 (1980), pp. 1251–1268.

[17] C. Fox, *A Conjugate Direction Sampler for Normal Distributions, with a Few Computed Examples*, Technical report 2008-1, Electronics Group, University of Otago, Dunedin, New Zealand, 2008.
[18] T. Gneiting, H. Sevcikova, D. B. Percival, M. Schlather, and Y. Jiang, *Fast and Exact Simulation of Large Gaussian Lattice Systems in $\mathbb{R}^2$: Exploring the Limits*, Technical report 477, Department of Statistics, University of Washington, Seattle, 2005.
[19] G. H. Golub, D. Ruiz, and A. Touhami, *A hybrid approach combining Chebyshev filter and conjugate gradient for solving linear systems with multiple right-hand sides*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 774–795.
[20] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., The Johns Hopkins University Press, Baltimore, 1996.
[21] J. Goodman and A. D. Sokal, *Multigrid Monte Carlo method. Conceptual foundations*, Phys. Rev. D, 40 (1989), pp. 2035–2071.
[22] D. Higdon, *A primer on space-time modelling from a Bayesian perspective*, in Statistical Methods for Spatio-Temporal Systems, B. Finkenstädt, L. Held, and V. Isham, eds., Chapman & Hall/CRC, Boca Raton, FL, 2007, pp. 217–279.
[23] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Nat. Bur. Standards, 45 (1950), pp. 255–282.
[24] C. Lanczos, *Solutions of linear equations by minimized iterations*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 33–53.
[25] D. J. C. MacKay, *Information Theory, Inference and Learning Algorithms*, Cambridge University Press, Cambridge, UK, 2003.
[26] G. Meurant, *The Lanczos and Conjugate Gradient Algorithms: From Theory to Finite Precision Computations*, SIAM, Philadelphia, 2006.
[27] R. B. Morgan, *Computing interior eigenvalues of large matrices*, Linear Algebra Appl., 154/156 (1991), pp. 289–309.
[28] R. B. Morgan, *Restarted block-GMRES with deflation of eigenvalues*, Appl. Numer. Math., 54 (2005), pp. 222–236.
[29] R. Neal, *Monte Carlo Implementation of Gaussian Process Models for Bayesian Regression and Classification*, Technical report 9702, Department of Statistics, University of Toronto, Toronto, 1997.
[30] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, New York, 2000.
[31] D. O'Leary, *Yet another polynomial preconditioner for the conjugate gradient algorithm*, Linear Algebra Appl., 154/156 (1991), pp. 377–388.
[32] B. N. Parlett, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, NJ, 1980.
[33] B. N. Parlett and I. S. Dhillon, *Relatively robust representations of symmetric tridiagonals*, Linear Algebra Appl., 309 (2000), pp. 121–151.
[34] C. E. Rasmussen, *Gaussian processes to speed up hybrid Monte Carlo for expensive Bayesian integrals*, in Bayesian Statistics 7, Oxford University Press, New York, 2003, pp. 651–660.
[35] A. C. Rencher, *Mulitvariate Statistical Inference and Applications*, Wiley, New York, 1998.
[36] G. O. Roberts and S. K. Sahu, *Updating schemes, correlation structure, blocking and parameterization for the Gibbs sampler*, J. Roy. Statist. Soc. Ser. B, 59 (1997), pp. 291–317.
[37] H. Rue, *Fast sampling of Gaussian Markov random fields*, J. R. Stat. Soc. Ser. B Stat. Methodol., 63 (2001), pp. 325–338.
[38] H. Rue and L. Held, *Gaussian Markov Random Fields: Theory and Applications*, Chapman & Hall/CRC, Boca Raton, FL, 2005.
[39] A. Ruhe, *Rational Krylov sequence methods for eigenvalue computation*, Linear Algebra Appl., 58 (1984), pp. 391–405.
[40] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, Manchester University Press, Manchester, UK, 1992.
[41] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, Philadelphia, 2003.
[42] J. A. Scales, *On the use of conjugate gradient to calculate the eigenvalues and singular values of large, sparse matrices*, Geophys. J. Int., 97 (1989), pp. 179–183.
[43] M. K. Schneider and A. S. Willsky, *Krylov subspace estimation*, SIAM J. Sci. Comput., 22 (2001), pp. 1840–1864.
[44] M. K. Schneider and A. S. Willsky, *Krylov subspace method for covariance approximation and random processes and fields*, Multidimens. Syst. Signal Process., 14 (2003), pp. 295–318.
[45] J. Shi and J. Malik, *Normalized cuts and image segmentation*, IEEE Trans. Pattern Anal. Mach. Intell., 22 (2000), pp. 888–905.
[46] H. D. Simon, *Analysis of the symmetric Lanczos algorithm with reorthogonalization methods*, Linear Algebra Appl., 61 (1984), pp. 101–131.

[47] G. L. C. Sleijpen and A. Van Der Sluis, *Further results on the convergence behavior of conjugate-gradients and Ritz values*, Linear Algebra Appl., 246 (1996), pp. 233–278.
[48] D. C. Sorensen, *Implicit application of polynomial filters in a k-step Arnoldi method*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 357–385.
[49] L. Tenorio, F. Andersson, M. de Hoop, and P. Ma, *Data analysis tools for uncertainty quantification of inverse problems*, Inverse Problems, 27 (2011), 045001.
[50] L. N. Trefethen and D. Bau, III, *Numerical Linear Algebra*, SIAM, Philadelphia, 1997.
[51] D. W. Vasco, L. R. Johnson, and O. Marques, *Global earth structure: Inference and assessment*, Geophys. J. Int., 137 (1999), pp. 381–407.
[52] W. Venables and B. Ripley, *Modern Applied Statistics with S-Plus*, 3rd ed., Springer-Verlag, New York, 1999.
[53] D. Watkins, *Fundamentals of Matrix Computations*, 2nd ed., Wiley, New York, 2002.
[54] C. K. Wikle, R. F. Milliff, D. Nychka, and M. Berliner, *Spatiotemporal hierarchical Bayesian modeling: Tropical ocean surface winds*, J. Amer. Statist. Assoc., 96 (2001), pp. 382–397.
[55] C. K. I. Williams and C. E. Rasmussen, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, MA, 2006.
[56] G. Wu, *A modified block Arnoldi algorithm with adaptive shifts for large interior eigenproblems*, J. Comput. Appl. Math., 205 (2007), pp. 343–363.