APPROXIMATION OF THE DETERMINANT OF LARGE SPARSE SYMMETRIC POSITIVE DEFINITE MATRICES

ARNOLD REUSKEN*

Abstract. This paper is concerned with the problem of approximating $\det(A)^{1/n}$ for a large sparse symmetric positive definite matrix A of order n. It is shown that an efficient solution of this problem is obtained by using a sparse approximate inverse of A. The method is explained and theoretical properties are discussed. A posteriori error estimation techniques are presented. Furthermore, results of numerical experiments are given which illustrate the performance of this new method.

Key words. determinant, sparse approximate inverse, preconditioning

AMS subject classifications. 65F10, 65F40, 65F50

1. Introduction. Throughout this paper, A denotes a real symmetric positive definite matrix of order n with eigenvalues

$$0 < \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n.$$

In a number of applications, for example in lattice Quantum Chromodynamics [12], certain functions of the determinant of A, such as $\det(A)^{1/n}$ or $\ln(\det(A))$ are of interest. It is well-known (cf. also §2) that for large n the function $A \rightarrow \det(A)$ has poor scaling properties and can be very ill-conditioned for certain matrices A. In this paper we consider the function

$$d: \ A \rightarrow \det(A)^{\frac{1}{n}} \ . \tag{1.1}$$

A few basic properties of this function are discussed in §2. In this paper we present a new method for approximating $d(A)$ for large sparse matrices A. The method is based on replacing A by a matrix which is in a certain sense close to A^{-1} and for which the determinant can be computed with low computational costs. One popular method for approximating A is based on the construction of an incomplete Cholesky factorization. This incomplete factorization is often used as a preconditioner when solving linear systems with matrix A. In this paper we use another preconditioning technique, namely that of sparse approximate inverses (cf. [1, 7, 9, 11]). In Remark 3.10 we comment on the advantages of the use of sparse approximate inverse preconditoning for approximating $d(A)$. Let $A = LL^T$ be the Cholesky decomposition of A. Then using techniques known from the literature a sparse approximate inverse G_E of L, i.e. a lower triangular matrix G_E which has a prescribed sparsity structure E and which is an approximation of L^{-1}, can be constructed. We then use $\det(G_E)^{-2/n} = \prod_{i=1}^{n} (G_E)_{ii}^{-2/n}$ as an approximation for $d(A)$. In §3 we explain the construction of G_E and discuss theoretical properties of this sparse approximate inverse. For example, such a sparse approximate inverse can be shown to exist for any symmetric positive definite A and has an interesting optimality property related to $d(A)$. As a direct consequence of this optimality property one obtains that $d(A) \leq \det(G_E)^{-2/n}$ holds and that the approximation of $d(A)$ by $\det(G_E)^{-2/n}$ becomes better if a larger sparsity pattern E is used.

*Institut für Geometrie und Praktische Mathematik, RWTH Aachen, Templergraben 55, D-52056 Aachen, Germany.

In §4 we consider the topic of error estimation. In the paper [2] bounds for the determinant of symmetric positive definite matrices are derived. These bounds, in which the Frobenius norm and an estimate of the extreme eigenvalues of the matrix involved are used, often yield rather poor estimates of the determinant (cf. experiments in [2]). In §4.1 we apply this technique to the preconditioned matrix $G_E A G_E^T$ and thus obtain reliable but rather pessimistic error bounds. It turns out that this error estimation technique is rather costly. In §4.2 we introduce a simple and cheap Monte Carlo technique for error estimation. In §5 we apply the new method to a few examples of large sparse symmetric positive definite matrices.

2. Preliminaries. In this section we discuss a few elementary properties of the function d. We give a comparision between the conditioning of the function d and of the fuction $A \to d(A)^n = \det(A)$. We use the notation $\|\cdot\|_2$ for the Euclidean norm and $\kappa(A) = \lambda_n/\lambda_1$ denotes the spectral condition number of A. The trace of the matrix A is denoted by $\mathrm{tr}(A)$.

LEMMA 2.1. *Let A and δA be symmetric positive definite matrices of order n. The following inequalities hold:*

$$\lambda_1 \leq d(A) \leq \lambda_n \ , \tag{2.1a}$$

$$d(A) \leq \frac{1}{n}\mathrm{tr}(A) \ , \tag{2.1b}$$

$$0 < \frac{d(A+\delta A) - d(A)}{d(A)} \leq \kappa(A)\frac{\|\delta A\|_2}{\|A\|_2} \ . \tag{2.1c}$$

Proof. The result in (2.1a) follows from

$$\lambda_1 \leq (\prod_{i=1}^{n} \lambda_i)^{\frac{1}{n}} \leq \lambda_n \ .$$

The result in (2.1b) follows from the inequality between the geometric and arithmetic mean:

$$d(A) = (\prod_{i=1}^{n} \lambda_i)^{\frac{1}{n}} \leq \frac{1}{n}\sum_{i=1}^{n} \lambda_i = \frac{1}{n}\mathrm{tr}(A) \ .$$

From the Courant-Fischer characterization of eigenvalues it follows that

$$\lambda_i(A+\delta A) \geq \lambda_i(A) + \lambda_1(\delta A) > \lambda_i(A)$$

for all i. Hence $d(A+\delta A) > d(A)$ holds. Now note that

$$\begin{aligned}
\frac{d(A+\delta A) - d(A)}{d(A)} &= \left(\det(I + A^{-1}\delta A)\right)^{\frac{1}{n}} - 1 \\
&= \left(\prod_{i=1}^{n}(1 + \lambda_i(A^{-1}\delta A))\right)^{\frac{1}{n}} - 1 \\
&\leq \left(\prod_{i=1}^{n}(1 + \|A^{-1}\|_2\|\delta A\|_2)\right)^{\frac{1}{n}} - 1 \\
&= \|A^{-1}\|_2\|\delta A\|_2 \ = \ \kappa(A)\frac{\|\delta A\|_2}{\|A\|_2} \ .
\end{aligned}$$

Thus the result in (2.1c) is proved. □

The result in (2.1c) shows that the function $d(A)$ is well-conditioned for matrices A which have a not too large condition number $\kappa(A)$.

We now briefly discuss the difference in conditioning between the functions $A \to d(A)$ and $A \to \det(A)$. For any symmetric positive definite matrix B of order n we have

$$d'(A)B := \lim_{t\to 0} \frac{d(A+tB)-d(A)}{t} = \frac{d(A)}{n}\mathrm{tr}(A^{-1}B) \ .$$

From the Courant-Fischer eigenvalue characterization we obtain $\lambda_i(A^{-1}B) \le \lambda_i(A^{-1})\|B\|_2$ for all i. Hence

$$\|d'(A)\|_2 := \max_{B \text{ is SPD}} \frac{|d'(A)B|}{\|B\|_2} = \frac{d(A)}{n} \max_{B \text{ is SPD}} \frac{\mathrm{tr}(A^{-1}B)}{\|B\|_2} \le \frac{d(A)}{n}\mathrm{tr}(A^{-1}) \ ,$$

with equality for $B = I$. Thus for the condition number of the function d we have

$$\frac{\|A\|_2\|d'(A)\|_2}{d(A)} = \frac{1}{n}\|A\|_2\mathrm{tr}(A^{-1}) \le \kappa(A) \ . \tag{2.2}$$

Note that for the diagonal matrix $A = \mathrm{diag}(A_{ii})$ with $A_{11} = 1$, $A_{ii} = \alpha \in (0,1)$ for $i > 1$, in the inequality in (2.2) one obtains equality for $n \to \infty$. For this A and with $\delta A = \varepsilon I$, $\varepsilon > 0$, for $n \to \infty$ we have equality in the second inequality in (2.1c), too.

For $\tilde{d}(A) = \det(A) = d(A)^n$ the condition number is given by

$$\frac{\|A\|_2\|\tilde{d}'(A)\|_2}{\tilde{d}(A)} = \frac{\|A\|_2 n d(A)^{n-1}\|d'(A)\|_2}{d(A)^n} = \|A\|_2\mathrm{tr}(A^{-1}) \ , \tag{2.3}$$

i.e. n times larger than the condition number in (2.2). The condition numbers for d and $\tilde{d}$ give an indication of the sensitivity if the perturbation $\|\delta A\|_2$ is sufficiently small. Note that the bound in (2.1c) is valid for arbitrary symmetric positive definite perturbations δA. The bound shows that even for larger perturbations the function $d(A)$ is well-conditioned at A if $\kappa(A)$ is not too large. For the function $\tilde{d}(A)$ the effect of relatively large perturbations can be much worse than for the asymptotic case ($\delta A \to 0$), which is characterized by the condition number in (2.3). Consider, for example, for $0 < \varepsilon < \frac{1}{2}$ a perturbation $\delta A = \varepsilon A$, i.e. $\|\delta A\|_2/\|A\|_2 = \varepsilon$. Then

$$\frac{\tilde{d}(A+\delta A)-\tilde{d}(A)}{\tilde{d}(A)} = (1+\varepsilon)^n - 1 \ge e^{\frac{1}{2}n\varepsilon} - 1 \ ,$$

which is very large if, for example, $\varepsilon = 10^{-3}$, $n = 10^5$.

The results in this section show that the numerical approximation of the function $d(A)$ is a much easier task than the numerical approximation of $A \to \det(A)$.

3. Sparse approximate inverse. In this section we explain and analyze the construction of a sparse approximate inverse of the matrix A. Let $A = LL^T$ be the Cholesky factorization of A, i.e. L is lower triangular and $L^{-1}AL^{-T} = I$. Note that $d(A) = d(L)^2 = \prod_{i=1}^n L_{ii}^{2/n}$. We will construct a sparse lower triangular approximation G of L^{-1} and approximate $d(A)$ by $d(G)^{-2} = \prod_{i=1}^n G_{ii}^{-2/n}$. The construction of a sparse approximate inverse that we use in this paper was introduced in [9, 10, 11] and can also be found in [1]. Some of the results derived in this section are presented in [1], too.

We first introduce some notation. Let $E \subset \{(i,j) \mid 1 \leq i,j \leq n\}$ be a given sparsity pattern. By $\#E$ we denote the number of elements in E. Let S_E be the set of $n \times n$ matrices for which all entries are set to zero if the corresponding index is *not* in E:

$$S_E = \{M \in \mathbb{R}^{n\times n} \mid M_{ij} = 0 \text{ if } (i,j) \notin E\} .$$

For $1 \leq i \leq n$ let $E_i = E \cap \{(i,j) \mid 1 \leq j \leq n\}$. If $n_i := \#E_i > 0$ we use the representation

$$E_i = \{(i,j_1),(i,j_2),\ldots,(i,j_{n_i})\}, \quad 1 \leq j_1 < j_2 < \ldots < j_{n_i} \leq n . \tag{3.1}$$

For $n_i > 0$ we define the projection

$$P_i : \ \mathbb{R}^n \to \mathbb{R}^{n_i} \ , \quad P_i(x_1,x_2,\ldots,x_n)^T = (x_{j_1},x_{j_2},\ldots,x_{j_{n_i}})^T . \tag{3.2}$$

Note that the matrix

$$P_i A P_i^T : \ \mathbb{R}^{n_i} \to \mathbb{R}^{n_i}$$

is symmetric positive definite. Typical choices of the sparsity pattern E (cf. §5) are such that n_i is a very small number compared to n (e.g. $n_i < 20$). In such a case the projected matrix $P_i A P_i^T$ has a small dimension.

To facilitate the analysis below, we first discuss the construction of an approximate sparse inverse $M_E \in S_E$ in a general framework. For $M_E \in S_E$ we use the representation

$$M_E = \begin{bmatrix} m_1^T \\ m_2^T \\ \vdots \\ m_n^T \end{bmatrix} , \quad m_i \in \mathbb{R}^n .$$

Note that if $n_i = 0$ then $m_i^T = (0,0,\ldots,0)$.

For given $A, B \in \mathbb{R}^{n\times n}$ with A symmetric positive definite we consider the following problem:

$$\text{determine } M_E \in S_E \text{ such that } (M_E A)_{ij} = B_{ij} \text{ for all } (i,j) \in E . \tag{3.3}$$

In (3.3) we have $\#E$ equations to determine $\#E$ entries in M_E. We first give two basic lemmas which will play an important role in the analysis of the sparse approximate inverse that will be defined in (3.9).

LEMMA 3.1. *The problem (3.3) has a unique solution $M_E \in S_E$. If $n_i > 0$ then the ith row of M_E is given by m_i^T with*

$$m_i = P_i^T (P_i A P_i^T)^{-1} P_i b_i , \tag{3.4}$$

where b_i^T is the ith row of B.

Proof. The equations in (3.3) can be represented as

$$(m_i^T A)_{j_k} = (b_i^T)_{j_k} \text{ for all } i \text{ with } n_i > 0 \text{ and all } k = 1,2,\ldots,n_i ,$$

where m_i^T is the ith row of M_E. Consider an i with $n_i > 0$. Note that $M_E \in S_E$, hence $P_i^T P_i m_i = m_i$. For the unknown entries in m_i we obtain the system of equations

$$(A P_i P_i^T m_i)_{j_k} = (b_i)_{j_k} , \quad k = 1,2,\ldots,n_i ,$$

which is equivalent to

$$P_i A P_i^T P_i m_i = P_i b_i \ .$$

The matrix $P_i A P_i^T$ is symmetric positive definite and thus m_i must satisfy

$$P_i m_i = (P_i A P_i^T)^{-1} P_i b_i \ .$$

Using $P_i^T P_i m_i = m_i$ we obtain the result in (3.4). The construction in this proof shows that the solution is unique. □

Below we use the Frobenius norm, denoted by $\|\cdot\|_F$:

$$\|B\|_F^2 = \sum_{i,j=1}^{n} B_{ij}^2 = \mathrm{tr}(BB^T) \ , \quad B \in \mathbb{R}^{n\times n}. \tag{3.5}$$

LEMMA 3.2. *Let $A = LL^T$ be the Cholesky factorization of A and let $M_E \in S_E$ be the unique solution of (3.3). Then M_E is the unique minimizer of the functional*

$$M \to \|(B - MA)L^{-T}\|_F^2 = \mathrm{tr}((B - MA)A^{-1}(B - MA)^T), \ M \in S_E. \tag{3.6}$$

Proof. Let e_i be the ith basis vector in $\mathbb{R}^n$. Take $M \in S_E$. The ith rows of M and B are denoted by m_i^T and b_i^T, respectively. Now note

$$\begin{aligned}\mathrm{tr}((B - MA)A^{-1}(B - MA)^T) &= \sum_{i=1}^{n} e_i^T (BA^{-1}B^T - MB^T - BM^T + MAM^T) e_i \\ &= \mathrm{tr}(BA^{-1}B^T) + \sum_{i=1}^{n} (-2m_i^T b_i + m_i^T A m_i) \ . \end{aligned} \tag{3.7}$$

The minimum of the functional (3.6) is obtained if in (3.7) we minimize the functionals

$$m_i \to -2m_i^T b_i + m_i^T A m_i \ , \quad m_i \in \mathcal{R}(P_i^T) \tag{3.8}$$

for all i with $n_i > 0$. If we write $m_i = P_i^T \hat{m}_i$, $\hat{m}_i \in \mathbb{R}^{n_i}$, then for $n_i > 0$ the functional (3.8) can be rewritten as

$$\hat{m}_i \to -2\hat{m}_i^T P_i b_i + \hat{m}_i^T P_i A P_i^T \hat{m}_i \ , \quad \hat{m}_i \in \mathbb{R}^{n_i} .$$

The unique minimum of this functional is obtained for $\hat{m}_i = (P_i A P_i^T)^{-1} P_i b_i$, i.e. $m_i = P_i^T (P_i A P_i^T)^{-1} P_i b_i$ for all i with $n_i > 0$. Using Lemma 3.1 it follows that M_E is the unique minimizer of the functional (3.6). □

Sparse approximate inverse. We now introduce the sparse approximate inverse that will be used as an approximation for L^{-1}. For this we choose a lower triangular pattern $E^l \subset \{(i,j) \mid 1 \le j \le i \le n\}$ and we assume that $(i,i) \in E^l$ for all i. The sparse approximate inverse is constructed in two steps:

$$1. \quad \hat{G}_{E^l} \in S_{E^l} \text{ such that } (\hat{G}_{E^l} A)_{ij} = \delta_{ij} \text{ for all } (i,j) \in E^l \ , \tag{3.9a}$$

$$2. \qquad G_{E^l} := (\mathrm{diag}(\hat{G}_{E^l}))^{-\frac{1}{2}} \hat{G}_{E^l} \ . \tag{3.9b}$$

The construction of G_{E^l} in (3.9) was first introduced in [9]. A theoretical background for this factorized sparse inverse is given in [11]. The approximate inverse $\hat{G}_{E^l}$ in (3.9a) is of the form (3.3) with $B = I$. From Lemma 3.1 it follows that in (3.9a) there is a unique solution $\hat{G}_{E^l}$. Note that because E^l is lower triangular and $(i,i) \in E^l$ we have $n_i = \#E^l > 0$ for all i and $j_{n_i} = i$ in (3.1). Hence it follows from Lemma 3.1 that the ith row of $\hat{G}_{E^l}$, denoted by g_i^T, is given by

$$\begin{aligned} g_i &= P_i^T (P_i A P_i^T)^{-1} P_i e_i, \qquad i = 1, 2, \ldots, n, \\ &= P_i^T (P_i A P_i^T)^{-1} \hat{e}_i, \quad \text{with } \hat{e}_i = (0, \ldots, 0, 1)^T \in \mathbb{R}^{n_i} . \end{aligned} \tag{3.10}$$

The ith entry of g_i, i.e. $e_i^T g_i$, is given by $\hat{e}_i^T (P_i A P_i^T)^{-1} \hat{e}_i$, which is strictly positive because $P_i A P_i^T$ is symmetric positive definite. Hence $\mathrm{diag}(\hat{G}_{E^l})$ contains only strictly positive entries and the second step (3.9b) is well-defined. Define $\hat{g}_i = P_i g_i$. The sparse approximate inverse $\hat{G}_{E^l}$ in (3.9a) can be computed by solving the (low dimensional) symmetric positive definite systems

$$P_i A P_i^T \hat{g}_i = (0, \ldots, 0, 1)^T, \ i = 1, 2, \ldots, n. \tag{3.11}$$

We now derive some interesting properties of the sparse approximate inverse as in (3.9). We start with a minimization property of $\hat{G}_{E^l}$:

THEOREM 3.3. *Let $A = LL^T$ be the Cholesky factorization of A and $D := \mathrm{diag}(L)$, $\hat{L} := LD$. $\hat{G}_{E^l}$ as in (3.9a) is the unique minimizer of the functional*

$$G \to \|(I - G\hat{L})D^{-1}\|_F^2 = \mathrm{tr}((I - G\hat{L})D^{-2}(I - G\hat{L})^T), \ G \in S_{E^l}. \tag{3.12}$$

Proof. The construction of $\hat{G}_{E^l}$ in (3.9a) is as in (3.3) with $E = E^l$, $B = I$. Hence Lemma 3.2 is applicable with $B = I$. It follows that $\hat{G}_{E^l}$ is the unique minimizer of

$$G \to \|(I - GA)L^{-T}\|_F^2 , \quad G \in S_{E^l} . \tag{3.13}$$

Decompose L^{-T} as $L^{-T} = D^{-1} + R$ with R strictly upper triangular. Then $D^{-1} - GL$ and R are lower and strictly upper triangular, respectively, and we obtain:

$$\begin{aligned} \|(I - GA)L^{-T}\|_F^2 &= \|(I - GLL^T)L^{-T}\|_F^2 = \|D^{-1} + R - GL\|_F^2 \\ &= \|D^{-1} - GL\|_F^2 + \|R\|_F^2 = \|(I - G\hat{L})D^{-1}\|_F^2 + \|R\|_F^2 . \end{aligned}$$

Hence the minimizers in (3.13) and (3.12) are the same. ∎

REMARK 3.4. From the result in Theorem 3.3 we see that in a scaled Frobenius norm (scaling with D^{-1}) $\hat{G}_{E^l}$ is the optimal approximation of $\hat{L}^{-1}$ in the set S_{E^l}, in the sense that $\hat{G}_{E^l}\hat{L}$ is closest to the identity. A seemingly more natural minimization problem is

$$\min_{G \in S_{E^l}} \|I - GL\|_F , \tag{3.14}$$

i.e. we directly approximate L^{-1} (instead of $\hat{L}^{-1}$) and do not use the scaling with D^{-1}. The minimization problem (3.14) is of the form as in Lemma 3.2 with $B = L^T$,

$E = E^l$. Hence the unique minimizer in (3.14), denoted by $\tilde{G}_{E^l}$, must satisfy (3.3) with $B = L^T$:

$$(\tilde{G}_{E^l} A)_{ij} = L_{ji} \quad \text{for all } (i,j) \in E^l \ . \tag{3.15}$$

Because E^l contains only indices (i,j) with $i \geq j$ and $L_{ji} = 0$ for $i > j$ it follows that $\tilde{G}_{E^l} \in S_{E^l}$ must satisfy

$$(\tilde{G}_{E^l} A)_{ij} = \begin{cases} 0 & \text{if } i \neq j \\ L_{ii} & \text{if } i = j \end{cases} \qquad \text{for all } (i,j) \in E^l \ . \tag{3.16}$$

This is similar to the system of equations in (3.9a), which characterizes $\hat{G}_{E^l}$. However, in (3.16) one needs the values L_{ii}, which in general are not available. Hence opposite to the minimization problem related to the functional (3.12) the minimization problem (3.14) is in general not solvable with acceptable computational costs. □

The following lemma will be used in the proof of Theorem 3.7.

LEMMA 3.5. *Let $\hat{G}_{E^l}$ be as in (3.9a). Decompose $\hat{G}_{E^l}$ as $\hat{G}_{E^l} = \hat{D}(I - \hat{L})$, with $\hat{D}$ diagonal and $\hat{L}$ strictly lower triangular. Define $E^l_- := E^l \setminus \{(i,i) \mid 1 \leq i \leq n\}$. Then $\hat{L}$ is the unique minimizer of the functional*

$$L \to \operatorname{tr}((I - L)A(I - L^T)) \ , \qquad L \in S_{E^l_-} \ , \tag{3.17}$$

and also of the functional

$$L \to \det[\operatorname{diag}((I - L)A(I - L^T))] \ , \qquad L \in S_{E^l_-} \ . \tag{3.18}$$

Furthermore, for $\hat{D}$ we have

$$\hat{D} = [\operatorname{diag}((I - \hat{L})A(I - \hat{L}^T))]^{-1} \ . \tag{3.19}$$

Proof. From the construction in (3.9a) it follows that

$$((I - \hat{L})A)_{ij} = 0 \quad \text{for all } (i,j) \in E^l_- \ ,$$

i.e., $\hat{L} \in S_{E^l_-}$ is such that $(\hat{L}A)_{ij} = A_{ij}$ for all $(i,j) \in S_{E^l_-}$. This is of the form (3.3) with $B = A$, $E = E^l_-$. From Lemma 3.2 we obtain that $\hat{L}$ is the unique minimizer of the functional

$$L \to \operatorname{tr}((A - LA)A^{-1}(A - LA)^T) = \operatorname{tr}((I - L)A(I - L^T)) \ , \quad L \in S_{E^l_-} \ ,$$

i.e., of the functional (3.17). From the proof of Lemma 3.2, with $B = A$, it follows that the minimization problem

$$\min_{L \in S_{E^l_-}} \operatorname{tr}((I - L)A(I - L^T))$$

decouples into seperate minimization problems (cf. (3.8)) for the rows of L:

$$\min_{l_i \in \mathcal{R}(P_i^T)} \{-2l_i^T a_i + l_i^T A l_i\} \tag{3.20}$$

for all i with $n_i > 0$. Here l_i^T and a_i^T are the ith rows of L and A, respectively. The minimization problem corresponding to (3.18) is

$$\min_{L\in S_{E^l_-}} \prod_{i=1}^{n}((I-L)A(I-L^T))_{ii} = \min_{L\in S_{E^l_-}} \prod_{i=1}^{n}(A_{ii} - 2l_i^T a_i + l_i^T A l_i) \ .$$

This decouples into the same minimization problems as in (3.20). Hence the functionals in (3.17) and (3.18) have the same minimizer.

Let $J = \mathrm{diag}((I-\hat{L})A(I-\hat{L}^T))$. Using the construction of $\hat{G}_{E^l}$ in (3.9a) we obtain

$$\begin{aligned}\hat{D}_{ii}^2 J_{ii} &= (\hat{D}(I-\hat{L})A(I-\hat{L}^T)\hat{D})_{ii} = (\hat{G}_{E^l}A\hat{G}_{E^l}^T)_{ii} \\ &= \sum_{k=1}^{n}(\hat{G}_{E^l}A)_{ik}(\hat{G}_{E^l})_{ik} = \sum_{\substack{k=1\\(i,k)\in E^l}}^{n} \delta_{ik}(\hat{G}_{E^l})_{ik} \\ &= (\hat{G}_{E^l})_{ii} = \hat{D}_{ii} \ .\end{aligned}$$

Hence $\hat{D}_{ii} = J_{ii}^{-1}$ holds for all i, i.e., (3.19) holds. ∎

COROLLARY 3.6. From (3.19) it follows that $\mathrm{diag}(\hat{G}_{E^l}A\hat{G}_{E^l}) = \mathrm{diag}(\hat{G}_{E^l})$ holds and thus, using (3.9b) we obtain

$$\mathrm{diag}(G_{E^l}AG_{E^l}) = I \tag{3.21}$$

for the sparse approximate inverse G_{E^l}. □

The following theorem gives a main result in the theory of approximate inverses. It was first derived in [11]. A proof can be found in [1], too.

THEOREM 3.7. *Let G_{E^l} be the approximate inverse in (3.9). Then G_{E^l} is the unique minimizer of the functional*

$$G \to \frac{\frac{1}{n}\mathrm{tr}(GAG^T)}{\det(GAG^T)^{\frac{1}{n}}} \ , \qquad G \in S_{E^l} \ . \tag{3.22}$$

Proof. For $G \in S_{E^l}$ we use the decomposition $G = D(I-L)$, with D diagonal and $L \in S_{E^l_-}$. Furthermore, for $L \in S_{E^l_-}$, $J_L := \mathrm{diag}((I-L)A(I-L^T))$. Now note

$$\begin{aligned}\frac{\frac{1}{n}\mathrm{tr}(GAG^T)}{\det(GAG^T)^{\frac{1}{n}}} &= \det(A)^{-\frac{1}{n}}\frac{\frac{1}{n}\mathrm{tr}((D(I-L)A(I-L^T)D)}{\det(G^2)^{\frac{1}{n}}} = \det(A)^{-\frac{1}{n}}\frac{\frac{1}{n}\mathrm{tr}(D^2J_L)}{\det(D^2)^{\frac{1}{n}}} \\ &= \det(A)^{-\frac{1}{n}}\frac{\frac{1}{n}\mathrm{tr}(D^2J_L)}{\det(D^2J_L)^{\frac{1}{n}}}\det(J_L)^{\frac{1}{n}} \geq \det(A)^{-\frac{1}{n}}\det(J_L)^{\frac{1}{n}} \ . \end{aligned} \tag{3.23}$$

The inequality in (3.23) follows from the inequality between the arithmetic and geometric mean: $\frac{1}{n}\sum_{i=1}^n \alpha_i \geq (\prod_{i=1}^n \alpha_i)^{1/n}$ for $\alpha_i \geq 0$.

For $\hat{G}_{E^l}$ in (3.9a) we use the decomposition $\hat{G}_{E^l} = \hat{D}(I-\hat{L})$. For the approximate inverse G_{E^l} we then have $G_{E^l} = (\mathrm{diag}(\hat{G}_{E^l}))^{-\frac{1}{2}}\hat{G}_{E^l} = \hat{D}^{\frac{1}{2}}(I-\hat{L})$. From Lemma 3.5 (3.18) it follows that $\det(J_L) \geq \det(J_{\hat{L}})$ for all $L \in S_{E^l_-}$. Furthermore from Lemma 3.5 (3.19) we obtain that for $G_{E^l} = \hat{D}^{\frac{1}{2}}(I-\hat{L})$ we have $(\hat{D}^{\frac{1}{2}})^2 J_{\hat{L}} = I$ and thus equality in (3.23) for $G = G_{E^l}$. We conclude that G_{E^l} is the unique minimizer of the functional

in (3.22). □

REMARK 3.8. The quantity

$$K(A) = \frac{\frac{1}{n}\mathrm{tr}(A)}{\det(A)^{\frac{1}{n}}}$$

can be seen as a nonstandard condition number (cf. [1, 9]). Properties of this quantity are given in [1] (Theorem 13.5). One elementary property is

$$1 \le K(A) \le \frac{\lambda_n}{\lambda_1} = \kappa(A) \ . \qquad \square$$

COROLLARY 3.9. For the approximate inverse G_{E^l} as in (3.9) we have (cf.(3.21))

$$1 \le K(G_{E^l} A G_{E^l}^T) = \frac{1}{\det(G_{E^l} A G_{E^l}^T)^{\frac{1}{n}}} \ ,$$

i.e.,

$$d(A) \le \det(G_{E^l}^2)^{-\frac{1}{n}} = \prod_{i=1}^{n} (G_{E^l})_{ii}^{-\frac{2}{n}} = \prod_{i=1}^{n} (\hat{G}_{E^l})_{ii}^{-\frac{1}{n}} \ . \tag{3.24}$$

Let $\tilde{E}^l$ be a lower triangular sparsity pattern that is larger than E^l, i.e., $E^l \subset \tilde{E}^l \subset \{(i,j) \mid 1 \le j \le i \le n\}$. From the optimality result in Theorem 3.7 it follows that

$$1 \le K(G_{\tilde{E}^l} A G_{\tilde{E}^l}^T) \le K(G_{E^l} A G_{E^l}^T) \ . \tag{3.25}$$

□ Motivated by the theoretical results in Corollary 3.9 we *propose to use the sparse approximate inverse* G_{E^l} as in (3.9) for approximating $d(A)$: Take $d(G_{E^l})^{-2} = d(\hat{G}_{E^l})^{-1}$ as an estimate for $d(A)$. Some properties of this method are discussed in the following remark.

REMARK 3.10. We consider the method of approximating $d(A)$ by $d(G_{E^l})^{-2} = d(\hat{G}_{E^l})^{-1}$. The practical realization of this method boils down to chosing a sparsity pattern E^l and solving the (small) systems in (3.11). We list a few properties of this approach:

1. The sparse approximate inverse exists for every symmetric positive definite A. Note that such an existence result does not hold for the incomplete Cholesky factorization. Furthermore, this factorization is obtained by solving low dimensional symmetric positive definite systems of the form $P_i A P_i^T \hat{g}_i = \hat{e}_i$ (cf. (3.11)), which can be realized in a stable way.

2. The systems $P_i A P_i^T \hat{g}_i = \hat{e}_i$, $i = 1, 2, \ldots, n$, can be solved in parallel.

3. For the computation of $d(G_{E^l})^{-2} = d(\hat{G}_{E^l})^{-1}$ we only need the diagonal entries of $\hat{G}_{E^l}$ (cf. (3.24)). In the systems $P_i A P_i^T \hat{g}_i = \hat{e}_i$ we then only have to compute the last entry of $\hat{g}_i$, i.e. $(\hat{g}_i)_{n_i}$. If these systems are solved using the Cholesky factorization, $P_i A P_i^T =: L_i L_i^T$ (L_i lower triangular) we only need the (n_i, n_i) entry of L_i, since $(\hat{g}_i)_{n_i} = (L_i)_{n_i n_i}^{-2}$.

4. The sparse approximate inverse has an optimality property related to the determinant: The functional $G \to K(GAG^T)$, $G \in S_{E^l}$, is minimal for G_{E^l}. From this the inequality (3.24) and the monotonicity result (3.25) follow.

5. From(3.24) we obtain the upper bound 0 for the relative error $d(A)/d(G_{E^l})^{-2}-1$. In §4 we will derive useful lower bounds for this relative error. These are a posteriori error bounds which use the matrix G_{E^l}. □

4. A posteriori error estimation. In the previous section it has been explained how an estimate $d(G_{E^l})^{-2}$ of $d(A)$ can be computed. From (3.24) we have the error bound

$$\frac{d(A)}{d(G_{E^l})^{-2}} \leq 1 \ . \tag{4.1}$$

In this section we will discuss a posteriori estimators for the error $d(A)/d(G_{E^l})^{-2}$. In §4.1 we apply the analysis from [2] to derive an a posteriori lower bound for the quantity in (4.1). This approach results in safe, but often rather pessimistic bounds for the error. In §4.2 we propose a very simple stochastic method for error estimation. This method, although it does not yield guaranteed bounds for the error, turns out to be very useful in practice.

4.1. Error estimation based on bounds from [2] . In this section we show how the analysis from [2] can be used to obtain an error estimator. We first recall a main result from [2] (Theorem 2). Let A be a symmetric positive matrix of order n, $\mu_1 = \mathrm{tr}(A)$, $\mu_2 = \|A\|_F^2$ and $\sigma(A) \subset [\alpha, \beta]$ with $\alpha > 0$. Then:

$$\begin{aligned} &\exp\left(\frac{1}{n}[\ln\alpha \ \ln t_l]\begin{bmatrix} \alpha & t_l \\ \alpha^2 & t_l^2 \end{bmatrix}^{-1}\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}\right) \leq d(A) \leq \\ &\exp\left(\frac{1}{n}[\ln\beta \ \ln t_u]\begin{bmatrix} \beta & t_u \\ \beta^2 & t_u^2 \end{bmatrix}^{-1}\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}\right) \ , \end{aligned} \tag{4.2}$$

where $t_l = \frac{\alpha\mu_1-\mu_2}{\alpha n-\mu_1}$, $t_u = \frac{\beta\mu_1-\mu_2}{\beta n-\mu_1}$.

In [2] this result is applied to obtain computable bounds for $d(A)$. Often these bounds yield rather poor estimates of $d(A)$. In the present paper we approximate $d(A)$ by $d(G_{E^l})^{-2}$ and use the result in (4.2) for *error estimation*. The upper bound (4.1) turns out to be satisfactory in numerical experiments (cf. §5). Therefore we restrict ourselves to the derivation of a lower bound for $d(A)/d(G_{E^l})^{-2}$, based on the left inequality in (4.2).

THEOREM 4.1. *Let G_{E^l} be the approximate inverse from (3.9) and $0 < \alpha \leq \lambda_{\min}(G_{E^l}AG_{E^l}^T)$, $\mu := \frac{1}{n}\|G_{E^l}AG_{E^l}^T\|_F^2$, $\delta := \mu - 1$. The following holds: $\alpha \leq 1$, $\delta \geq 0$ and*

$$\exp\left(\frac{1}{(\alpha-1)^2+\delta}\left(\delta\ln\alpha + (1-\alpha)^2\ln(1+\frac{\delta}{1-\alpha})\right)\right) \leq \frac{d(A)}{d(G_{E^l})^{-2}} \leq 1 \ . \tag{4.3}$$

Proof. The right inequality in (4.3) is already given in (4.1). We introduce the notation $\tau_1 \leq \tau_2 \leq \ldots \leq \tau_n$ for the eigenvalues of $G_{E^l}AG_{E^l}^T$. From (3.21) we obtain $\frac{1}{n}\sum_{i=1}^n \tau_i = 1$ and from this it follows that $\alpha \leq \tau_1 \leq 1$ holds. Furthermore,

$$1 = (\frac{1}{n}\sum_{i=1}^n \tau_i)^2 \leq \frac{1}{n}\sum_{i=1}^n \tau_i^2$$

yields $\mu = \frac{1}{n}\mathrm{tr}((G_{E^l} A G_{E^l}^T)^2) \geq 1$ and thus $\delta \geq 0$. We now use the left inequality in (4.2) applied to the matrix $G_{E^l} A G_{E^l}^T$. Note that

$$\mu_1 = \mathrm{tr}(G_{E^l} A G_{E^l}^T) = n,\ \mu_2 = n\mu,\ t_l = \frac{\alpha\mu_1 - \mu_2}{\alpha n - \mu_1} = 1 + \frac{\delta}{1-\alpha}\ .$$

A simple computation yields

$$\frac{1}{n}\begin{bmatrix} \alpha & t_l \\ \alpha^2 & t_l^2 \end{bmatrix}^{-1} \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} = \frac{1}{t_l - \alpha}\begin{bmatrix} \frac{\delta}{1-\alpha} \\ 1-\alpha \end{bmatrix}\ , \tag{4.4}$$

and

$$t_l - \alpha = \frac{(1-\alpha)^2 + \delta}{1-\alpha}\ . \tag{4.5}$$

Substitution of (4.5) in (4.4) results in

$$\begin{aligned}\frac{1}{n}[\ln\alpha\ \ \ln t_l]\begin{bmatrix} \alpha & t_l \\ \alpha^2 & t_l^2 \end{bmatrix}^{-1} \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} &= \frac{1}{(1-\alpha)^2+\delta}\left(\delta\ln\alpha + (1-\alpha)^2\ln t_l\right) \\ &= \frac{1}{(1-\alpha)^2+\delta}\left(\delta\ln\alpha + (1-\alpha)^2\ln(1+\frac{\delta}{1-\alpha})\right)\ .\end{aligned}$$

Using this the left inequality in (4.3) follows from the left inequality in (4.2). □

Note that for the lower bound in (4.3) to be computable, we need $\mu = \frac{1}{n}\|G_{E^l} A G_{E^l}^T\|_F^2$ and a strictly positive lower bound α for the smallest eigenvalue of $G_{E^l} A G_{E^l}^T$. We now discuss methods for computing α and μ. These methods are used in the numerical experiments in §5.

We first discuss two methods for computing α. The first method, which can be applied if A is an M-matrix, is based on the following lemma, where we use the notation $\mathbf{1} = (1, 1, \ldots, 1)^T \in \mathbb{R}^n$.

LEMMA 4.2. *Let A be a symmetric positive definite matrix of order n with $A_{ij} \leq 0$ for all $i \neq j$ and G_{E^l} its sparse approximate inverse (3.9). Furthermore, let z be such that*

$$\|G_{E^l} A G_{E^l}^T z - \mathbf{1}\|_\infty \leq \eta < 1\ .$$

Then

$$(1-\eta)\|z\|_\infty^{-1} \leq \lambda_{\min}(G_{E^l} A G_{E^l}^T) \tag{4.6}$$

holds.

Proof. From the assumptions it follows that A is an M-matrix. In [11] (Theorem 4.1) it is proved that then $G_{E^l} A G_{E^l}^T$ is an M-matrix, too. Let $z^* = (G_{E^l} A G_{E^l}^T)^{-1}\mathbf{1}$. Because $(G_{E^l} A G_{E^l}^T)^{-1}$ has only nonnegative entries it follows that

$$\begin{aligned}\|(G_{E^l} A G_{E^l}^T)^{-1}\|_\infty = \|z^*\|_\infty &= \|z + (z^* - z)\|_\infty \\ &\leq \|z\|_\infty + \|(G_{E^l} A G_{E^l}^T)^{-1}\|_\infty \|G_{E^l} A G_{E^l}^T z - \mathbf{1}\|_\infty \\ &\leq \|z\|_\infty + \|(G_{E^l} A G_{E^l}^T)^{-1}\|_\infty \eta\ .\end{aligned}$$

Hence $\|(G_{E^l}AG_{E^l}^T)^{-1}\|_\infty^{-1} \geq (1-\eta)\|z\|_\infty^{-1}$. Using $\lambda_{\min}(G_{E^l}AG_{E^l}^T) \geq \|((G_{E^l}AG_{E^l}^T)^{-1}\|_\infty^{-1}$ we obtain the result (4.6). □

Based on this lemma we obtain the following method for computing α. Choose $0 < \eta \ll 1$ and apply the conjugate gradient method to the system $G_{E^l}AG_{E^l}^T z^* = \mathbf{1}$. This results in approximations $z^0, z^1, \ldots$ of z^*. One iterates until the stopping criterion $d^j := \|G_{E^l}AG_{E^l}^T z^j - \mathbf{1}\|_\infty \leq \eta$ is satisfied. Then take $\alpha := (1-d^j)\|z^j\|_\infty^{-1}$. In view of efficiency one should not take a very small tolerance η. In our experiments in §5 we use $\eta = 0.2$ and $z^0 = \mathbf{1}$. Note that the CG method is applied to a system with the *preconditioned* matrix $G_{E^l}AG_{E^l}^T$. In situations where the preconditioning is effective one may expect that relatively few CG iterations are needed to compute z^j such that $\|G_{E^l}AG_{E^l}^T z^j - \mathbf{1}\|_\infty \leq \eta$ is satisfied. Results of numerical experiments are presented in §5.

As a second method for determining α, which is applicable to any symmetric positive definite A, we propose the Lanczos method for approximating eigenvalues applied to the matrix $G_{E^l}AG_{E^l}^T$. This method yields a decreasing sequence $\lambda_1^{(1)} \geq \lambda_1^{(2)} \geq \cdots \geq \lambda_1^{(j)} \geq \lambda_{\min}(G_{E^l}AG_{E^l}^T)$ of approximations $\lambda_1^{(j)}$ of $\lambda_{\min}(G_{E^l}AG_{E^l}^T)$. If

$$\lambda_1^{(j)} - \lambda_{\min}(G_{E^l}AG_{E^l}^T) < \varepsilon \tag{4.7}$$

holds, then $\alpha = \lambda_1^{(j)} - \varepsilon$ can be used in Theorem 4.1. However, in practice it is usually not known how to obtain reasonable values for ε in (4.7). Therefore, in our experiments we use a simple heuristic for error estimation (instead of a rigorous bound as in (4.7)), based on the observed convergence behaviour of $\lambda_1^{(j)}$ (cf. §5).
It is known that for the Lanczos method the convergence to extreme eigenvalues is relatively fast. Moreover, it often occurs that the small eigenvalues of the preconditioned matrix $G_{E^l}AG_{E^l}^T$ are well-separated from the rest of the spectrum, which has a positive effect on the convergence speed $\lambda_1^{(j)} \to \lambda_{\min}(G_{E^l}AG_{E^l}^T)$. In numerical experiments we indeed observe that often already after a few Lanczos iterations we have an approximation of $\lambda_{\min}(G_{E^l}AG_{E^l}^T)$ with an estimated relative error of a few percent. However, for the α computed in this second method we do not have a rigorous analysis which guarantees that $0 < \alpha < \lambda_{\min}(G_{E^l}AG_{E^l}^T)$ holds. Nevertheless, from numerical experiments we see that this method is satisfactory. This is partly explained by the relatively fast convergence of the Lanczos method towards the smallest eigenvalue. A further explanation follows from the form of the lower bound in (4.3). For $\alpha \ll 1$, $\delta \ll 1$, which is typically the case in our experiments in §5, this lower bound essentially behaves like $\exp(\delta \ln \alpha) =: g(\alpha)$. Note that $0 < \frac{g'(\alpha)\alpha}{g(\alpha)} = \delta \ll 1$ holds. Hence the sensitivity of the lower bound with respect to perturbations in α is very mild.

We now discuss the computation of the quantity $\mu = \frac{1}{n}\|G_{E^l}AG_{E^l}^T\|_F^2$, which is needed in (4.3). Clearly, for computing μ one needs the matrices G_{E^l} and A. To avoid unnecessary storage requirements one should not compute the matrix $X := G_{E^l}AG_{E^l}^T$ and then determine $\mu = \frac{1}{n}\|X\|_F^2$. A with respect to storage more efficient approach can be based on

$$\|G_{E^l}AG_{E^l}^T\|_F^2 = \sum_{i=1}^{n} \|G_{E^l}AG_{E^l}^T e_i\|_2^2 \ ,$$

where e_i is the ith basis vector in $\mathbb{R}^n$. For the computation of $\|G_{E^l}AG_{E^l}^T e_i\|_2^2$,

$i = 1, 2, \ldots, n$, which can be done in parallel, one needs only sparse matrix-vector multiplications with the matrices G_{E^l} and A. Furthermore, for the computation of $AG_{E^l}^T e_i$ one can use that $(DG_{E^l}A)_{ij} = (\hat{G}_{E^l}A)_{ij} = \delta_{ij}$ for $(i,j) \in E^l$, with $D := \mathrm{diag}(G_{E^l})$ (cf. (3.9)). It follows from (3.10) that

$$\begin{aligned} AG_{E^l}^T e_i &= (I - P_i^T P_i) AG_{E^l}^T e_i + P_i^T P_i AG_{E^l}^T e_i \\ &= (I - P_i^T P_i) AG_{E^l}^T e_i + P_i^T P_i A\hat{G}_{E^l}^T D^{-1} e_i \\ &= (I - P_i^T P_i) AG_{E^l}^T e_i + D_{ii}^{-1} e_i \end{aligned}$$

holds.

REMARK 4.3. Note that for the error estimators discussed in this section the matrix G_{E^l} must be available (and thus stored), whereas for the computation of the approximation $d(G_{E^l})^{-2}$ of $d(A)$ we do not have to store the matrix G_{E^l} (cf. Remark 3.10 item 3). Furthermore, as we will see in §5, the computation of these error estimators is relatively expensive. □

4.2. Error estimation based on a Monte Carlo approach. In this section we discuss a simple error estimation method which turns out to be useful in practice. Opposite to those treated in the previous section this method does not yield (an approximation of) bounds for the error.
The exact error is given by

$$\frac{d(A)}{d(G_{E^l})^{-2}} = d(G_{E^l} A G_{E^l}^T) = d(\mathcal{E}_{E^l}) \ ,$$

where $\mathcal{E}_{E^l} := G_{E^l} A G_{E^l}^T$ is a sparse symmetric positive definite matrix. Fore ease of presentation we assume that the pattern E^l is sufficiently large such that

$$\rho(I - \mathcal{E}_{E^l}) < 1 \tag{4.8}$$

holds. In [11] it is proved that if A is an M-matrix or a (block) H-matrix then (4.8) is satisfied for every lower triangular pattern E^l. In the numerical experiments (cf. §5) with matrices which are not M-matrices or (block) H-matrices (4.8) turns out to be satisfied for standard choices of E^l. We note that if (4.8) does not hold then the technique discussed below can still be applied if one replaces $\mathcal{E}_{E^l}$ by $\omega\mathcal{E}_{E^l}$ with $\omega > 0$ a suitable damping factor such that $\rho(I - \omega\mathcal{E}_{E^l}) < 1$ is satisfied.

For the exact error we obtain, using a Taylor expansion of $\ln(I-B)$ for $B \in \mathbb{R}^{n\times n}$ with $\rho(B) < 1$ (cf. [6]):

$$\begin{aligned} d(\mathcal{E}_{E^l}) &= \exp\left(\frac{1}{n}\ln(\det(\mathcal{E}_{E^l}))\right) = \exp\left(\frac{1}{n}\mathrm{tr}(\ln(\mathcal{E}_{E^l}))\right) \\ &= \exp\left(\frac{1}{n}\mathrm{tr}(\ln(I - (I - \mathcal{E}_{E^l})))\right) = \exp\left(-\frac{1}{n}\mathrm{tr}(\sum_{k=1}^{\infty}\frac{(I-\mathcal{E}_{E^l})^k}{k})\right) \end{aligned} \tag{4.9}$$

Hence, an error estimation can be based on estimates for the partial sums $S_m := \sum_{k=1}^{m}\frac{1}{k}\mathrm{tr}((I-\mathcal{E}_{E^l})^k)$. The construction of G_{E^l} is such that $\mathrm{diag}(\mathcal{E}_{E^l}) = I$ (cf. (3.21)) and thus $\mathrm{tr}(\mathcal{E}_{E^l}) = n$ and $S_1 = 0$. For S_2 we have

$$S_2 = \frac{1}{2}\mathrm{tr}((I - \mathcal{E}_{E^l})^2) = \frac{1}{2}\mathrm{tr}(I - 2\mathcal{E}_{E^l} + \mathcal{E}_{E^l}^2) = -\frac{1}{2}n + \frac{1}{2}\mathrm{tr}(\mathcal{E}_{E^l}^2) \ . \tag{4.10}$$

For S_3 we obtain

$$S_3 = \frac{1}{2}\mathrm{tr}((I-\mathcal{E}_{E^l})^2) + \frac{1}{3}\mathrm{tr}((I-\mathcal{E}_{E^l})^3) = -\frac{7}{6}n + \frac{3}{2}\mathrm{tr}(\mathcal{E}_{E^l}^2) - \frac{1}{3}\mathrm{tr}(\mathcal{E}_{E^l}^3) \ . \tag{4.11}$$

Note that in S_2 and S_3 the quantity $\mathrm{tr}(\mathcal{E}_{E^l}^2) = \|\mathcal{E}_{E^l}\|_F^2 = \|G_{E^l} A G_{E^l}^T\|_F^2$ occurs which is also used in the error estimator in §4.1. In this section we use a *Monte Carlo method* to approximate the trace quantities in S_m. The method we use is based on the following proposition [8, 3].

PROPOSITION 4.4. *Let H be a symmetric matrix of order n with $\mathrm{tr}(H) \neq 0$. Let V be the discrete random variable which takes the values 1 and -1 each with probability 0.5 and let z be a vector of n independent samples from V. Then $z^T H z$ is an unbiased estimator of $\mathrm{tr}(H)$:*

$$E(z^T H z) = \mathrm{tr}(H) \ ,$$

and

$$var(z^T H z) = 2\sum_{i \neq j} h_{ij}^2 \ .$$

For approximating the trace quantity in S_2 we use the following Monte Carlo algorithm:

for $j = 1, 2, \ldots, M$

1. Generate $z_j \in \mathbb{R}^n$ with entries which are uniformly distributed in $(0,1)$.
2. If $(z_j)_i < 0.5$ then $(z_j)_i := -1$, otherwise, $(z_j)_i := 1$.
3. $y_j := \mathcal{E}_{E^l} z_j, \quad \alpha_j := y_j^T y_j$.

Based on Proposition 4.4 and (4.10) we use

$$\hat{S}_2 := -\frac{1}{2}n + \frac{1}{2M}\sum_{j=1}^{M} \alpha_j \tag{4.12}$$

as an approximation for S_2. The corresponding error estimator is

$$E_2 = \exp(-\frac{1}{n}\hat{S}_2). \tag{4.13}$$

For the approximation of S_3 we replace step 3 in the algorithm above by

3. $y_j := \mathcal{E}_{E^l} z_j, \quad \hat{y}_j := \mathcal{E}_{E^l} y_j, \quad \alpha_j := \frac{3}{2} y_j^T y_j - \frac{1}{3} y_j^T \hat{y}_j$

and we use

$$\hat{S}_3 := -\frac{7}{6}n + \frac{1}{M}\sum_{j=1}^{M} \alpha_j \tag{4.14}$$

as an estimate for S_3. The corresponding error estimator is

$$E_3 = \exp(-\frac{1}{n}\hat{S}_3) \ . \tag{4.15}$$

Clearly, this technique can be extended to the partial sums S_m with $m > 3$. However, in our applications we only use $\hat{S}_2$ and $\hat{S}_3$ for error estimation. It turns out that, at least in our experiments, the two leading terms in the expansion (4.9) are sufficient for a reasonable error estimation. Note that due to the truncation of the Taylor expansion, the estimators E_2 and E_3 for $d(\mathcal{E}_{E^l})$ are biased.

It is shown in [3] that based on the so-called Hoeffding inequality (cf. [13]) probabilistic bounds for $|\frac{1}{M}\sum_{i=1}^{M} z_i^T H z_i - \mathrm{tr}(H)|$ can be derived, where $z_1, z_2, \ldots, z_M$ are independent random variables as in Proposition 4.4. In this paper we do not use these bounds. Based on numerical experiments we take a fixed small value for the parameter M in the Monte Carlo algorithm above (in the experiments in §5: $M = 6$).

REMARK 4.5. In the setting of this paper Proposition 4.4 is applied with $H = p(\mathcal{E}_{E^l})$, where p is a known polynomial of degree 2 or 3. In the Monte Carlo technique for approximating $\det(A) = \exp(\mathrm{tr}(\ln(A)))$ from [3], Proposition 4.4 is applied with $H = \ln(A)$. The quantity $z^T \ln(A) z$, which can be considered as a Riemann-Stieltjes integral, is approximated using suitable quadrature rules. In [3] this quadrature is based on a Gauss-Christoffel technique where the unknown nodes and weights in the quadrature rule are determined using the Lanczos method. For a detailed explanation of this method we refer to [3].
A further alternative that could be considered for error estimation is the use of this method from [3]. In the setting here, this method could be used to compute a (rough) approximation of $\det(G_{E^l} A G_{E^l}^T)^{1/n}$. We did not investigate this possibility. The results in [2, 3] give an indication that this alternative is probably much more expensive than the method presented in this section. □

5. Numerical experiments. In this section we present some results of numerical experiments with the methods introduced in §3 and §4. All experiments are done using a MATLAB implementation. We use the MATLAB notation $nnz(B)$ for the number of nonzero entries in a matrix B.

Experiment 1 (discrete 2D Laplacian). We consider the standard 5–point discrete Laplacian on a uniform square grid with m mesh points in both directions, i.e. $n = m^2$. For this symmetric positive definite matrix the eigenvalues are known:

$$\lambda_{\nu\mu} = 4(m+1)^2 \left(\sin^2(\frac{\nu\pi}{2(m+1)}) + \sin^2(\frac{\mu\pi}{2(m+1)}) \right), \quad 1 \le \nu, \mu \le m \ .$$

For the choice of the sparsity pattern E^l we use a simple approach based on the nonzero structure of (powers of) the matrix A:

$$E^l(k) := \{(i,j) \mid i \ge j \text{ and } (A^k)_{ij} \neq 0\} \ , \quad k = 1, 2, \ldots . \tag{5.1}$$

We first describe some features of the methods for the case $m = 30$, $k = 2$ and after that we will vary m and k. Let A denote the discrete Laplacian for the case $m = 30$ and L_A its lower triangular part. We then have $nnz(L_A) = 2640$. For the sparse approximate inverse we obtain $nnz(G_{E^l(2)}) = 6002$. The systems $P_i A P_i^T \hat{g}_i = (0, 0, \ldots, 0, 1)^T$ that have to be solved to determine $G_{E^l(2)}$ (cf. (3.11)) have dimensions between 1 and 7; the mean of these dimensions is 6.7. As an approximation of $d(A) = 3.1379\ 10^3$ we obtain

$$d(G_{E^l(2)})^{-2} = d(\hat{G}_{E^l(2)})^{-1} = \prod_{i=1}^{n} (\hat{G}_{E^l(2)})_{ii}^{-\frac{1}{n}} = 3.2526\ 10^3 \ .$$

Hence $d(A)/d(G_{E^l(2)})^{-2} = 0.965$. For the computation of this approximation along the lines as described in Remark 3.10, item 3, we have to compute the Cholesky factorizations $P_i A P_i^T = L_i L_i^T$, $i = 1, 2, \ldots, n$. For this approximately 41 10^3 flops are needed (in the MATLAB implementation). If we compare this with the costs of one matrix–vector multiplication $A * x$ (8760 flops), denoted by MATVEC, it follows that for computing this approximation of $d(A)$, with an error of 3.5 percent, we need arithmetic work comparable to only 5 MATVEC.

We will see that the arithmetic costs for error estimation are significantly higher. We first consider the methods of §4.1. The arithmetic costs are measured in terms of MATVEC. For the computation of α as indicated in Lemma 4.2 with $\eta = 0.2$, using the CG method with starting vector $\mathbf{1} = (1, 1, \ldots, 1)^T$ we need 8 iterations. In each CG iteration we have to compute a matrix–vector multiplication $G_{E^l} A G_{E^l}^T x$, which costs approximately 3.7 MATVEC. We obtain $\alpha_{\mathrm{CG}} = 0.0155$. For the method based on the Lanczos method for approximating $\lambda_{\min}(G_{E^l} A G_{E^l}^T)$ we use the heuristic stopping criterion

$$|\lambda_1^{(j)} - \lambda_1^{(j-1)}| < 0.01 |\lambda_1^{(j)}| \,. \tag{5.2}$$

We then need 7 Lanczos iterations, resulting in $\alpha_{\mathrm{Lanczos}} = 0.0254$. A direct computation results in $\lambda_{\min}(G_{E^l} A G_{E^l}^T) = 0.025347$.
For the computation of $\mu = \|G_{E^l} A G_{E^l}^T\|_F^2$ we first computed the lower triangular part of $X = G_{E^l} A G_{E^l}^T$ and then computed $\|X\|_F$ (making use of symmetry). The total costs of this are approximately 18 MATVEC. Application of Lemma 4.1, with α_{CG} and $\alpha_{\mathrm{Lanczos}}$ yields the two intervals

$$[0.880, 1] \quad \text{and} \quad [0.895, 1] \,,$$

which both contain the exact error 0.965. In both cases, the total costs for error estimation are 40–45 MATVEC, which is approximately 10 times more than the costs for computing the approximation $d(G_{E^l(2)})^{-2}$.

We now consider the method of §4.2. We use the estimators E_2 and E_3 from (4.13), (4.15) with $M = 6$. The results are $E_2 = 0.980$, $E_3 = 0.973$. Note that the order of magnitude of the exact error (3.5 percent) is approximated well by both E_2 (2.0 percent) and E_3 (2.7 percent). In step 3 in the Monte Carlo algorithm for computing $\hat{S}_2$ we need one matrix–vector multiplication $G_{E^l} A G_{E^l}^T x$ (costs 3.7 MATVEC). The total arithmetic costs for E_2 are approximately 20 MATVEC. For $\hat{S}_3$ we need two matrix–vector multiplications with $\mathcal{E}_{E^l}$ in the third step of the Monte Carlo algorithm. The total costs for E_3 are approximately 40 MATVEC.

In Table 5.1 we give results for the discrete 2D Laplacian with $m = 30$ ($n = 900$), $m = 100$ ($n = 10000$) and $m = 200$ ($n = 40000$). We use the sparsity pattern $E^l(2)$. In the third column of this table we give the computed approximation of $d(A)$ and the corresponding relative error. In the fourth column we give the total arithmetic costs for the Cholesky factorization of the matrices $P_i A P_i^T$, $i = 1, 2, \ldots, n$ (cf. Remark 3.10, item 3). In the columns 5–8 we give the results and corresponding arithmetic costs for the error estimators discussed in §4. The fifth column corresponds to the method discussed in §4.1 with α determined using the CG method applied to $G_{E^l} A G_{E^l}^T = \mathbf{1}$ with starting vector $\mathbf{1}$. In the stopping criterion we take $\eta = 0.2$ (cf. Lemma 4.2). The computed $\alpha = \alpha_{\mathrm{CG}}$ is used as input for the lower bound in (4.3). The resulting bound for the relative error and the arithmetic costs for computing this error bound are shown in column 5. In column 6 one finds the computed error bounds if α is determined using the Lanczos method with stopping criterion (5.2). In the last two

TABLE 5.1
Results for 2D discrete Laplacian with $E^l = E^l(2)$

n	$d(A)$	$d(G_{E^l(2)})^{-2}$ (error)	costs for $d(G_{E^l(2)})^{-2}$	Thm. 4.1, α_{CG}	Thm. 4.1, $\alpha_{\mathrm{Lanczos}}$	MC E_2	MC E_3
900	$3.138\ 10^3$	$3.253\ 10^3$ (3.5%)	5 MV	$\leq$ 12% (45 MV)	$\leq$ 11% (45 MV)	2.0% (20 MV)	2.7% (40 MV)
10000	$3.292\ 10^4$	$3.434\ 10^4$ (4.1%)	5 MV	$\leq$ 21% (140 MV)	$\leq$ 19% (102 MV)	2.2% (24 MV)	2.6% (48 MV)
40000	$1.300\ 10^5$	$1.359\ 10^5$ (4.3%)	5 MV	$\leq$ 26% (276 MV)	$\leq$ 24% (159 MV)	2.2% (24 MV)	2.7% (48 MV)

TABLE 5.2
Results for 2D discrete Laplacian with $E^l = E^l(4)$

n	$d(A)$	$d(G_{E^l(4)})^{-2}$ (error)	costs for $d(G_{E^l(4)})^{-2}$	Thm. 4.1, α_{CG}	Thm. 4.1, $\alpha_{\mathrm{Lanczos}}$	MC E_2	MC E_3
900	$3.138\ 10^3$	$3.177\ 10^3$ (1.2%)	41 MV	$\leq$ 3.5% (137 MV)	$\leq$ 3.0% (146 MV)	0.65% (54 MV)	1.1% (108 MV)
10000	$3.292\ 10^4$	$3.347\ 10^4$ (1.6%)	45 MV	$\leq$ 7.7% (263 MV)	$\leq$ 7.0% (226 MV)	0.91% (55 MV)	1.1% (110 MV)
40000	$1.300\ 10^5$	$1.323\ 10^5$ (1.7%)	46 MV	$\leq$ 10% (487 MV)	$\leq$ 9.3% (348 MV)	0.93% (56 MV)	1.1% (112 MV)

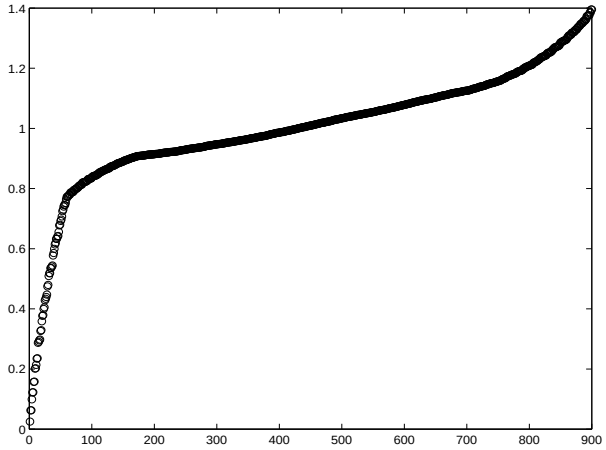
columns the results for the Monte Carlo estimators E_2 (4.13) and E_3 (4.15) are given. In Table 5.2 we show the results and corresponding arithmetic costs for the method with sparsity pattern $E^l = E^l(4)$.

Concerning the numerical results we note the following. From the third and fourth column in Table 5.1 we see that using this method we can obtain an approximation of $d(A)$ with relative error only a few percent and arithmetic costs only a few MATVEC. Moreover, this efficiency hardly depends on the dimension n. Comparison of the third and fourth columns of the Tables 5.1 and 5.2 shows that the approximation significantly improves if we enlarge the pattern from $E^l(2)$ to $E^l(4)$. The corresponding arithmetic costs increase by a factor of about 9. This is caused by the fact that the mean of the dimensions of the systems $P_i A P_i^T$, $i = 1, 2, \ldots, n$, increases from approximately 7 ($E^l(2)$) to approximately 20. For $n = 10000$ we have $nnz(L_A) = 29800$, $nnz(G_{E^l(2)}) = 69002$, $nnz(G_{E^l(4)}) = 204030$. For the other n values we have similar ratios between the number of nonzeros in the matrices L_A and G_{E^l}. Note that the matrix G_{E^l} has to be stored for the error estimation but not for the computation of the approximation $d(G_{E^l})^{-2}$. The error bounds in the fifth and sixth column in the Tables 5.1 and 5.2 are rather conservative and expensive. Furthermore there is some deterioration in the quality and a quite strong increase in the costs if the dimension n grows. The strong increase in the costs is mainly due to the fact that the CG and Lanczos method both need significantly more iterations if n increases. This is a well-known phenomenom (the matrix $G_{E^l} A G_{E^l}^T$ has a condition number that is proportional to n). Also note that the costs for these error estimators are (very) high compared to the costs of the computation of $d(G_{E^l})^{-2}$. The results in the last two columns indicate that the Monte Carlo error estimators, although less reliable, are more favourable.

In Figure 5.1 we show the eigenvalues of the matrix $G_{E^l} A G_{E^l}^T$ for the case $n = 900$, $E^l = E^l(2)$ (computed with the MATLAB function EIG). The eigenvalues are in the

interval $[0.025, 1.4]$. The mean of these eigenvalues is 1 ($\mathrm{tr}(G_{E^l} A G_{E^l}^T) = 1$). One can see that relatively many eigenvalues are close to 1 and only a few eigenvalues are close to zero.

FIG. 5.1. *Eigenvalues of the matrix* $G_{E^l} A G_{E^l}^T$ *in Experiment 1*



Experiment 2 (MATLAB random sparse matrix). The sparsity structure of the matrices considered in Experiment 1 is very regular. In this experiment we consider matrices with a pattern of nonzero entries that is very irregular. We used the MATLAB generator (SPRAND$(n, n, 2/n)$) to generate a matrix B of order n with approximately $2n$ nonzero entries. These are uniformly distributed random entries in $(0, 1)$. The matrix $B^T B$ is then sparse symmetric positive semidefinite. In the generic case this matrix has many eigenvalues zero. To obtain a positive definite matrix we generated a random vector d with all entries chosen from a uniform distribution on the interval $(0, 1)$ ($d :=$RAND$(n, 1)$). As a testmatrix we used $A := B^T B + \mathrm{diag}(d)$. We performed numerical experiments similar to those in Experiment 1 above. We only consider the case with sparsity pattern $E^l = E^l(2)$. The error estimator based on the CG method is not applicable because the sign condition in Lemma 4.2 is not fulfilled. For the case $n = 900$ the eigenvalues of A and of $G_{E^l} A G_{E^l}^T$ are shown in Figure 5.2. For A the smallest and largest eigenvalues are 0.0099 and 5.70, respectively. The picture on the right in Figure 5.2 shows that for this matrix A sparse approximate inverse preconditioning results in a very well–conditioned matrix. Related to this, one can see in Table 5.3 that for this random matrix A the approximation of $d(A)$ based on the sparse approximate inverse is much better than for the discrete Laplacian in Experiment 1. For $n = 900, 10000, 40000$ we obtain $nnz(L_A) = 2730, 29789, 120216$ and $nnz(G_{E^l}) = 7477, 82290, 335139$, respectively. For $n = 900, 10000, 40000$ the mean of the dimensions of the systems $P_i A P_i^T$, $i = 1, 2, \ldots, n$ is $10.6, 10.8, 11.0$, respectively. In all three cases the costs for a matrix–vector multiplication $G_{E^l} A G_{E^l} x$ are approximately 4.3 MV. Furthermore, in all three cases the matrix $G_{E^l} A G_{E^l}^T$ is well–conditioned and the number of Lanczos iterations needed to satisfy the stopping criterion (5.2) hardly depends on n. Due to this, for increasing n, the growth in the costs for the error estimator based on Theorem 4.1 (column 5) is much slower than in Experiment 1. As in the Tables 5.1 and 5.2, in Table 5.3 the error quantities in the columns 3, 5,6,7 are bounds or estimates for the relative error $1 - d(G_{E^l} A G_{E^l}^T)$.

FIG. 5.2. *Eigenvalues of the matrices A and $G_{E^l} A G_{E^l}^T$ in Experiment 2*

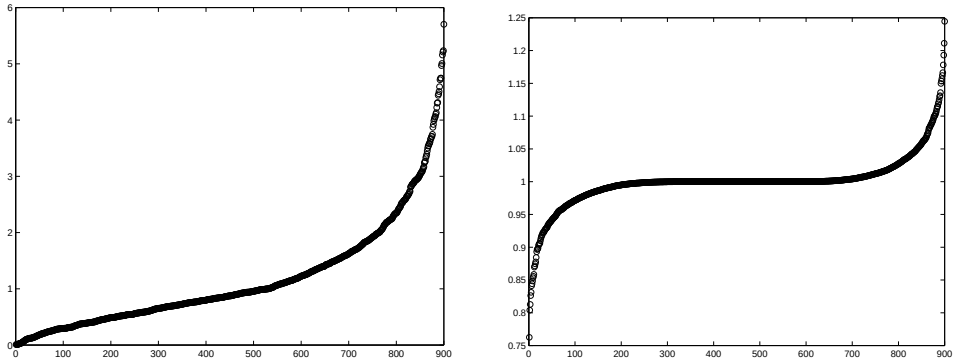


TABLE 5.3
Results for MATLAB random sparse matrices with $E^l = E^l(2)$

n	$d(A)$	$d(G_{E^l})^{-2}$ (error)	costs for $d(G_{E^l})^{-2}$	Thm. 4.1, α_{Lanczos}	MC E_2	MC E_3
900	0.82453	0.82521 ($8.3\ 10^{-4}$)	23 MV	$\leq 9.8\ 10^{-4}$ (110 MV)	$1.4\ 10^{-3}$ (26 MV)	$1.0\ 1.0^{-3}$ (52 MV)
10000	–	0.81053 (–)	18 MV	$\leq 1.1\ 10^{-3}$ (139 MV)	$8.4\ 10^{-4}$ (26 MV)	$7.4\ 10^{-4}$ (52 MV)
40000	–	0.82033 (–)	18 MV	$\leq 1.0\ 10^{-3}$ (146 MV)	$6.2\ 10^{-4}$ (26 MV)	$8.3\ 10^{-4}$ (52 MV)

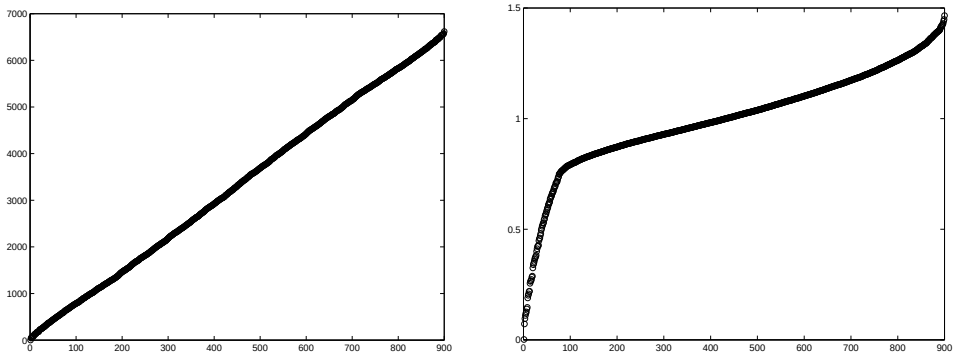
For $n = 10000, 40000$ the values of $d(A)$ are not given (column 2). This has to do with the fact that for these matrices with very irregular sparsity patterns the Cholesky factorization $A = LL^T$ suffers from much more fill-in than for the matrices in the Experiments 1 and 3. For the matrix A in this experiment with $n = 900$ we have $nnz(L_A) = 2730$ and $nnz(L) = 72766$. For $n = 10000$ we run into storage problems if we try to compute the Cholesky factorization using the MATLAB function CHOL.

Experiment 3 (QCD type matrix). In this experiment we consider a complex Hermitean positive definite matrix with sparsity structure as in Experiment 1. This matrix is motivated by applications from the QCD field. In QCD simulations the determinant of the so-called Wilson fermion matrix is of interest. These matrices and some of their properties are discussed in [4, 5]. The nonzero entries in a Wilson fermion matrix are induced by a nearest neighbour coupling in a regular 4-dimensional grid. These couplings consist of 12×12 complex matrices M_{xy}, which have a tensor product structure $M_{xy} = P_{xy} \otimes U_{xy}$, where $P_{xy} \in \mathbb{R}^{4\times 4}$ is a projector, $U_{xy} \in \mathbb{C}^{3\times 3}$ is from SU_3 and x and y denote nearest neighbours in the grid. These coupling matrices M_{xy} strongly fluctuate as a function of x and y. Here we consider a (toy) problem with a matrix which has some similarities with these Wilson fermion matrices. We start with a 2-dimensional regular grid as in Experiment 1 (n grid points). For the couplings with nearest neighbours we use complex numbers with length 1. These numbers are chosen as follows. The couplings with south and west neighbours at a grid point x are $\exp(2i\pi\alpha_S(x))$ and $\exp(2i\pi\alpha_W(x))$, respectively, where $\alpha_S(x)$ and $\alpha_W(x)$ are chosen from a uniform distribution on the interval $(0, 1)$. The couplings with the north and east neighbours are taken such that the matrix is hermitean. To make the comparison with Experiment 1 easier the matrix is scaled by the factor n,

i.e. the couplings with nearest neighbours have length n. For the diagonal we take γI, where γ is chosen such that the smallest eigenvalue of the resulting matrix is approximately 1 (this can be realized by using the MATLAB function EIGS for estimating the smallest eigenvalue). We performed numerical experiments as in Experiment 1 with $E^l = E^l(2)$. The number of nonzero entries in L_A and G_{E^l} are the same as in Experiment 1. For $n = 900$ the eigenvalues of the matrices A and $G_{E^l} A G_{E^l}^T$ are shown in Figure 5.3. These spectra are in the intervals $[1,\ 6.6\,10^3]$ and $[1.7\,10^{-3},\ 1.5]$, respectively.

The results of numerical experiments are presented in Table 5.4. Note that the error

FIG. 5.3. *Eigenvalues of the matrices A and $G_{E^l} A G_{E^l}^T$ in Experiment 3*



estimator from §4.1 in which the CG method is used for computing α can not be used for this matrix (assumptions in Lemma 4.2 are not satisfied). We did not consider the case $n = 40000$ here because then the application of the EIG function for computing the smallest eigenvalue led to memory problems.

Comparison of the results in Table 5.4 with those in Table 5.1 shows that when the

TABLE 5.4
Results for QCD type matrix with $E^l = E^l(2)$

n	$d(A)$	$d(G_{E^l})^{-2}$ (error)	costs for $d(G_{E^l})^{-2}$	Thm. 4.1, α_{Lanczos}	MC E_2	MC E_3
900	$2.500\ 10^3$	$2.620\ 10^3$ (4.6%)	5 MV	$\leq 24\%$ (79 MV)	2.6% (23 MV)	3.3% (46 MV)
10000	$2.739\ 10^4$	$2.842\ 10^4$ (3.6%)	5 MV	$\leq 31\%$ (133 MV)	2.4% (23 MV)	2.7% (46 MV)
22500	$6.173\ 10^4$	$6.391\ 10^4$ (3.4%)	5 MV	$\leq 32\%$ (198 MV)	2.3% (23 MV)	2.8% (46 MV)

method is applied to the QCD type of problem instead of the discrete Laplacian the performance of the method does not change very much.

Finally, we note that in all measurements of the arithmetic costs we did not take into account the costs of determining the sparsity pattern $E^l(k)$ and of building the matrices $P_i A P_i^T$.

REFERENCES

[1] O. AXELSSON, *Iterative Solution Methods*, Cambridge University Press, New York, 1994.
[2] Z. BAI AND G. H. GOLUB, *Bounds on the trace of the inverse and the determinant of symmetric positive definite matrices*, Annals of Numer. Math., 4 (1997), pp. 29–38.

[3] Z. Bai, M. Fahey and G. H. Golub, *Some large scale matrix computation problems*, J. Comput. Appl. Math., 74 (1996), pp. 71–89.

[4] P. de Forcrand, *Progress on lattice QCD algorithms*, Nuclear Physics B (Proc. Suppl.), 47 (1996), pp. 228–235.

[5] A. Frommer and B. Medeke, *Exploiting structure in Krylov subspace methods for the Wilson fermion matrix*, in 15th IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics, A. Sydow (ed.), Wissenschaft & Technik Verlag, Berlin (1997), pp. 485–490.

[6] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Second ed., The Johns Hopkins University Press, Baltimore, MD, 1989.

[7] M. J. Grote and T. Huckle, *Parallel preconditioning with sparse approximate inverses*, SIAM J. Sci. Comput., 18 (1997), pp. 838–853.

[8] M. Hutchinson, *A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines*, Commun. Statist. -Simula., 18 (1989), pp. 1059–1076.

[9] I. E. Kaporin, *An alternative approach to estimating the convergence rate of the CG method*, In Numerical Methods and Software, Yu. A. Kuznetsov, ed., Dept. of Numerical Mathematics, USSR Academy of Sciences, Moscow, 1990, pp. 55–72. (In Russian.)

[10] L. .Yu. Kolotilina and A. Yu. Yeremin, *On a family of two-level preconditionings of the incomplete block factorization type*, Soviet J. Numer. Anal. Math. Model., 1 (1986), pp. 293–320.

[11] L. .Yu. Kolotilina and A. Yu. Yeremin, *Factorized sparse approximate inverse preconditionings I : Theory*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 45–58.

[12] I. Montvay and G. Münster, *Quantum Fields on a Lattice*, Cambridge University Press, Cambridge, 1994.

[13] D. Pollard, *Convergence of Stochastic Processes*, Springer, New York, 1984.