Conditioning analysis of incomplete Cholesky factorizations with orthogonal dropping

Artem Napov*

Computational Research Division,
Lawrence Berkeley National Laboratory (MS 50F-1148),
One Cyclotron Rd.
Berkeley, CA 94720, USA.

Report LBNL-5353E

March 2012
Revised April 2012

Abstract

The analysis of preconditioners based on incomplete Cholesky factorization in which the neglected (dropped) components are orthogonal to the approximations being kept is presented. General estimate for the condition number of the preconditioned system is given which only depends on the accuracy of individual approximations. The estimate is further improved if, for instance, only the newly computed rows of the factor are modified during each approximation step. In this latter case it is further shown to be sharp. The analysis is illustrated with some existing factorizations in the context of discretized elliptic partial differential equations.

Key words. incomplete Cholesky, conditioning analysis, convergence analysis, iterative methods, preconditioner

AMS subject classification. 65F08, 65F35

*This work was supported by the Director, Office of Science, Office of Advanced Scientific Computing Research of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

1 Introduction

We consider incomplete Cholesky factorizations for the iterative solution of symmetric positive definite (SPD) $N \times N$ linear systems

$$A\mathbf{u} = \mathbf{b}\,. \tag{1.1}$$

Incomplete Cholesky factorizations are commonly described with the help of Cholesky version of Gaussian elimination, which amounts to compute an upper triangular matrix R such that $A = R^T R$. Incomplete factorization is then obtained by introducing approximation into the elimination process (see, e.g., [1, 14, 22, 24]).

Here, we are concerned with incomplete factorizations based on low-rank approximation. These techniques have been successfully applied to linear systems arising from discretized partial differential equations (PDEs) [3, 4, 13, 26, 25, 27, 16]. Unlike classical incomplete factorization methods [20, 17, 21], which rely on the dropping of individual entries, the new methods approximate some dense parts of the factors with low-rank matrices. The resulting approximate factors may then remain dense but acquire some structure, and the term *data-sparse* is often used to describe them. It is now known that for systems arising from discretization of PDEs, individual low-rank approximations are often possible with almost arbitrary accuracy for a rank which is independent of, or slowly varying with, the block size [4, 13, 9] (see also [5, 6] for the related results) . Whereas substantial effort have been invested in finding the applications where the low-rank property is present, the impact of the individual approximations on the quality of the preconditioner is less well understood.

In this paper, we present the analysis which covers two such preconditioners: [16] and, under a slightly modified form, [27]. Both methods are variants of incomplete Cholesky factorization. Whereas they are conceptually different and produce factors in different data-sparse formats, they both exploit the low-rank property by "dropping" components which are orthogonal to the low-rank approximations being kept. This is motivated by the observation that the successively formed Schur complements then do not decrease (in the SPD sense) when approximation is performed, which in turn guarantees that the preconditioner can be constructed for any SPD A.

The analysis in the present paper also applies to any SPD A. The starting point is a generic incomplete Cholesky factorization algorithm which forms a common framework for the aforementioned factorizations. It allows approximation on a large part of the factor and also requiers the orthogonality between the components rejected during each approximation stage and the approximation that are kept.

The resulting factorization is then considered as a preconditioner to the original system. We present a general upper bound on its condition number (i.e., the quotient of its largest and smallest eigenvalues), as required to estimate the convergence rate of iterative methods such as conjugate gradient [1, 14, 22]. The bound involves quantities which only depend on the individual orthogonal approximations and in a way measure their *accuracy*.

Now, the algorithms in [16] and [27] modify different groups of rows during the approximation steps. This may further lead to an improved condition estimate involving the same accuracy measures if the modified rows (and the rows with lower indices) are no longer modified for the few following steps. The ideal case when only the newly computed rows are modified at each step (and, hence, each row is modified at most once) is of particular interest. This case is referred to as *one-level* for reasons which are made clear below. The related estimate is shown to be sharp in that, for every set of accuracy values, there exist a matrix and a corresponding incomplete factorization preconditioner, obtained using orthogonal dropping, that allow to reach the bound.

We note that a related accuracy measure has been introduced in [27]. The analysis there is however applied in a rather model case when only one approximation step is performed. In this context, for instance, no distinction can be made between a general and a "localized" dropping mentioned above, as the bounds in both cases lead to the same condition number estimate.

Another particular case arises when the approximated blocks of the factor are set to zero, and the resulting preconditioner corresponds to a block-diagonal part of A. The accuracy measures then coincide with so-called CBS constants, which are commonly used in the eigenvalue estimates of a block-diagonally preconditioned system. Again, the case when the preconditioner has 2×2 blockdiagonal form is well understood [2, 1]; however, the extension to block-diagonal preconditioners with multiple blocks which arise form our analysis leads to a better bound than one could obtain by recursively applying the 2×2 estimate. In addition, the above-mentioned sharpness property carries over to the block-diagonal case (and, for simplicity, we only prove this property for block-diagonal preconditioners).

Eventually, our analysis is illustrated with the factorization algorithms in [27], [16] and the one-level variant in the context of model problem arising from a low-order discretization of a second-order PDE. Numerical experiments reveal that all the considered preconditioners have similar conditioning properties, and further, that the bound for the one-level variant allows an accurate prediction of their condition number. On the other hand, based on the analysis, the approximation schemes are modified to keep the condition number bounded independently of the problem size; their effectiveness is also assessed in the model problem context.

The reminder of the paper is organized as follows. In Section 2 we introduce our generic incomplete Cholesky factorization with orthogonal dropping and relate it to the existing methods. The bounds on the condition number are presented in

Section 3 and illustrated on a model problem in Section 4. Concluding remarks are stated in Section 5.

Notation

$[i,j] = \{i, i+1, ..., j\}$ stands for the ordered set of integers ranging from i to j. I stands for identity matrix and O for zero matrix (matrix with all entries being zero).

For any vector $\mathbf{v}$, $\|\mathbf{v}\|$ is its Euclidian norm. For any matrix C, the induced matrix norm is

$$\|C\| = \max_{\mathbf{v}\neq\mathbf{0}} \frac{\|C\mathbf{v}\|}{\|\mathbf{v}\|} .$$

For any SPD matrix D, $\lambda_{\max}(D)$ and $\lambda_{\min}(D)$ is, respectively, its largest and its smallest eigenvalue. Since $\lambda_{\min}(D) > 0$, the spectral condition number $\kappa(D) = \lambda_{\max}(D)/\lambda_{\min}(D)$ is well defined. For any $n \times n$ block matrix $E = (E_{i,j})$ and any $1 \leq i \leq k \leq n$,

$$E_{i:k,j} = \begin{pmatrix} E_{i,j}^T & \cdots & E_{k,j}^T \end{pmatrix}^T ,$$

and, for any $1 \leq j \leq m \leq n$,

$$E_{i:k,j:m} = \begin{pmatrix} E_{i:k,j} & \cdots & E_{i:k,m} \end{pmatrix} .$$

2 Factorization algorithm

2.1 General setting

Let the index set $[1, N]$ be partitioned into n ($n > 0$) disjoint contiguous subsets $\mathcal{I}_i$, $i = 1, ..., n$. The corresponding block partitioning of A is given by

$$A = \begin{pmatrix} A_{1,1} & \cdots & A_{1,n} \\ \vdots & \ddots & \vdots \\ A_{n,1} & \cdots & A_{n,n} \end{pmatrix}, \tag{2.1}$$

where $A_{i,j}$ is a $|\mathcal{I}_i| \times |\mathcal{I}_j|$ matrix and, since A is symmetric, $A_{i,j} = A_{j,i}^T$, $i, j = 1, ..., n$.

Before we introduce the incomplete factorization algorithm, we briefly recall with Algorithm 2.1 a block variant of the (exact) Cholesky factorization as applies to the above setting. It computes an upper triangular factor

$$R = \begin{pmatrix} R_{1,1} & \cdots & R_{1,n} \\ & \ddots & \vdots \\ & & R_{n,n} \end{pmatrix}, \tag{2.2}$$

such that $A = R^T R$, where $R_{i,j}$ is a $|\mathcal{I}_i| \times |\mathcal{I}_j|$ matrix, $R_{i,i}$ is upper triangular and, for $i > j$, $R_{i,j} = O$, $i,j = 1,...,n$. During the step i, the ith block row of the factor R is computed by first adding to the ith block row of A the corresponding contributions from the already computed rows of R (line 1a), then factorizing the pivot block (line 1b), and eventually forming the corresponding block row (line 1c).

```
Algorithm 2.1 (Block Cholesky) : $R = \textsc{BkChol}(A)$
1. for $i = 1,...,n$
1a.     if $(i = 1)$ : $R_{1,1:n} \leftarrow A_{1,1:n}$
        else          $R_{i,i:n} \leftarrow A_{i,i:n} - R^T_{1:i-1,i} R_{1:i-1,i:n}$
1b.     Compute an upper triangular $R_S$ such that $R_S^T R_S = R_{i,i}$ ;
1c.     $R_{i,i} \leftarrow R_S$
        $R_{i,i+1:n} \leftarrow R_S^{-T} R_{i,i+1:n}$
```

The first i block rows of the factor may be shown to satisfy (see e.g., [1, 12])

$$A = \begin{pmatrix} R^T_{1:i,1:i} & \\ R^T_{1:i,i+1:n} & S_A^{(i)} \end{pmatrix} \begin{pmatrix} R_{1:i,1:i} & R_{1:i,i+1:n} \\ & I \end{pmatrix}, \tag{2.3}$$

where

$$S_A^{(i)} = A_{i+1:n,i+1:n} - R^T_{1:i,i+1:n} R_{1:i,i+1:n}$$

is the Schur complement of A corresponding to the bottom right $(n-i) \times (n-i)$ block.

Now, the incomplete Cholesky factorization is given by Algorithm 2.2. It performs ℓ approximation steps ($\ell \geq n$). Prior to the step k, some n_{k-1} block rows of the factor have already been computed, with $n > n_\ell \geq \ldots \geq n_1 > n_0 = 0$, and n_k block rows are available at the end of this step. Note that several approximation steps may be performed without computing new rows of the factor, in which case the corresponding n_k are equal. Without loss of generality, we further assume that $n = n_\ell + 1$ and either $n_k = n_{k-1} + 1$ or $n_k = n_{k-1}$, $k = 1,...,\ell$.

Computing new rows of the factor (lines 1a-1c) is now supplemented with an approximation (dropping) stage (line 1d). Because of this latter, the factorization is no longer exact; it (implicitly) generates a sequence of "approximations" B_k of A, $k = 1,...,\ell$, where

$$B_k = \begin{pmatrix} {R_{11}^{(k)}}^T & \\ {\widetilde{R}_{12}^{(k)}}^T & \widetilde{S}_B^{(k)} \end{pmatrix} \begin{pmatrix} R_{11}^{(k)} & \widetilde{R}_{12}^{(k)} \\ & I \end{pmatrix}, \tag{2.4}$$

with $R_{11}^{(k)} = R_{1:n_k,1:n_k}$, $R_{12}^{(k)} = R_{1:n_k,n_k+1:n}$ being the first n_k block rows of the factor at the end of step k and with

$$\widetilde{S}_B^{(k)} = A_{n_k+1:n,n_k+1:n} - {\widetilde{R}_{12}^{(k)}}^T \widetilde{R}_{12}^{(k)} \tag{2.5}$$

denoting the corresponding Schur complement of B_k. By implicitly we mean that the product (2.4) is not explicitly formed; for $k < \ell$, it corresponds to an intermediate factorization at step k, whereas $B_\ell = R^T R$ is the resulting preconditioner. Eventually, since $n_\ell < n$, the algorithm completes the factorization at line 2.

Algorithm 2.2 (Incomplete (block) Cholesky) : $R \;=\; \textsc{IBkChol}(A)$

```
1.  for $k = 1, \dots, \ell$
        if $(n_k \;>\; n_{k-1})$ :
1a.         if $(k \;=\; 1)$ : $R_{1,1:n} \;\leftarrow\; A_{1,1:n}$
            else     $R_{n_k,n_k:n} \;\leftarrow\; A_{n_k,n_k:n} - R^T_{1:n_k-1,n_k} R_{1:n_k-1,n_k:n}$
1b.         Compute an upper triangular $R_S$ such that $R_S^T R_S = R_{n_k,n_k}$ ;
1c.         $R_{n_k,n_k} \;\leftarrow\; R_S$
            $R_{n_k,n_k+1:n} \;\leftarrow\; R_S^{-T} R_{n_k,n_k+1:n}$
1d.     $R_{1:n_k,n_k+1:n} \;\leftarrow\; \text{APPROX}_k\left(R_{1:n_k,n_k+1:n}\right)$ .

2.  Compute an upper triangular $R_{n,n}$ such that $R^T_{n,n} R_{n,n} = A_{n,n} - R^T_{1:n,n} R_{1:n,n}$ ;
```

Note that, similarly to the exact factorization, the lines 1a-1c do not alter the current approximation of A; that is, the preconditioner implicitly available at the end of step $k-1$ is the same as the one available prior to line 1d during the step k. Hence, setting $B_0 = A$, one further has

$$B_{k-1} \;=\; \begin{pmatrix} {R_{11}^{(k)}}^T & \\ {R_{12}^{(k)}}^T & S_B^{(k)} \end{pmatrix} \begin{pmatrix} R_{11}^{(k)} & R_{12}^{(k)} \\ & I \end{pmatrix}, \qquad k = 1, \dots, \ell\,, \tag{2.6}$$

where $R_{12}^{(k)} \;=\; R_{1:n_k,n_k+1:n}$ now corresponds to the rows of the factor prior to the approximation stage (line 1d) of step k, and where

$$S_B^{(k)} \;=\; A_{n_k+1:n,n_k+1:n} - {R_{12}^{(k)}}^T R_{12}^{(k)} \;. \tag{2.7}$$

One further sees from the line 1d that

$$\widetilde{R}_{12}^{(k)} \;=\; \text{APPROX}_k\left(R_{12}^{(k)}\right) . \tag{2.8}$$

Regarding this operation, we note that our analysis does not rely on any of its particular implementations. In what follows, we mainly assume that the dropped component is orthogonal to the one that is kept; that is, we mainly require

$${\widetilde{R}_{12}^{(k)}}^T \left(R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} \right) \;=\; O \qquad \forall\ \widetilde{R}_{12}^{(k)} \;=\; \text{APPROX}_k\left(R_{12}^{(k)}\right) \tag{2.9}$$

to hold. The only additional assumption we make is on the indices of block rows in $R_{12}^{(k)}$ that are effectively modified by $\textsc{approx}_k(\cdot)$ operation. Some examples of approximation operations that fulfill the above orthogonality condition are discussed in Section 2.2 below.

Now, as proved in Section 3, the assumption (2.9) further implies that the Algorithm 2.2 always terminates and that the intermediate matrices B_k, $k = 1, ..., \ell - 1$, as well as the final preconditioner $B_\ell = R^T R$ are SPD. Hence, the condition number

$$\kappa(R^{-T}AR^{-1}) \;=\; \frac{\lambda_{\max}(R^{-T}AR^{-1})}{\lambda_{\min}(R^{-T}AR^{-1})} \tag{2.10}$$

of the preconditioned system is well defined.

2.2 Orthogonal dropping

The $\textsc{approx}(\cdot)$ operation is commonly implemented using a truncated version of an orthogonal decomposition, such as truncated SVD or or rank-revealing QR [7, 8, 10, 15]. For a given threshold tol_a, it produces a factorization

$$R_{12}^{(k)} \;=\; \begin{pmatrix} Q_1 & Q_2 \end{pmatrix} \begin{pmatrix} U_1 & U_2 \end{pmatrix}^T , \tag{2.11}$$

where $Q = \begin{pmatrix} Q_1 & Q_2 \end{pmatrix}$ is orthogonal and

$$\|U_2\| \;\leq\; tol_a \, .$$

The approximation then corresponds to a rank deficient (also called low-rank) term $\widetilde{R}_{12}^{(k)} = Q_1 U_1^T$ whose rank r_k is given by the number of columns in Q_1. On the other hand, the truncation error is given by $R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} = Q_2 U_2^T$. Hence, the condition (2.9) follows directly from $Q_1^T Q_2 = O$, whereas the orthogonality of columns in Q_2 further implies

$$\| \, R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} \, \| \;\leq\; tol_a \, . \tag{2.12}$$

In the case of truncated SVD the threshold may be chosen explicitly, by discarding the singular values that are lower than the threshold value. This holds for an absolute truncation threshold tol_a, but also for a relative one, which amounts to $tol_a = tol_r \| \, R_{12}^{(k)} \, \|$, since $\| \, R_{12}^{(k)} \, \|$ then corresponds to the largest singular value. Regarding the rank revealing factorizations, the threshold is only available indirectly, usually trough an inequality like $tol_r \, , \, tol_a \;<\; p \cdot tol_{\mathrm{RRQR}}$, where tol_{RRQR} is the truncation threshold of the rank-revealing algorithm and p is a low order polynomial depending on the dimensions of $R_{12}^{(k)}$ [10, 15].

Now, in practice the approximation is usually not applied to the whole block $R_{12}^{(k)}$. For instance, if only a few rows of $R_{12}^{(k)}$ need to be modified, the decomposition (2.11) is applied to those rows only. One then has

$$R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} = \Pi \begin{pmatrix} O \\ R^{(k)} - \widetilde{R}^{(k)} \end{pmatrix} ,$$

where $R^{(k)}$ corresponds to the rows of $R_{12}^{(k)}$ which are approximated by $\widetilde{R}^{(k)}$, and Π is a permutation which enumerates those rows last. The "local" condition

$$\widetilde{R}^{(k)\,T} (R^{(k)} - \widetilde{R}^{(k)}) = O$$

is then easily shown to imply (2.9), and, since Π is orthogonal, one has

$$\| \, R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} \, \| = \| R^{(k)} - \widetilde{R}^{(k)} \| \, .$$

2.3 Relation to existing methods

The incomplete Cholesky factorization described in Algorithm 2.2 provides a suitable framework that covers several existing preconditioners. This is, for instance, the case of incomplete Cholesky factorizations in [16, 27] (the latter being considered here under a slightly different form, see below). Both methods are similar in that they exploit individual low-rank approximations to reduce both the storage requirement and the operation complexity of the factorization. More precisely, they require at most $\mathcal{O}(r_{\max} N^2)$ operations to factorize a $N \times N$ matrix, where $r_{\max} = \max_{1 \le k \le \ell} r_k$ is the maximal rank from the approximation step, and need at most $\mathcal{O}(r_{\max} N)$ storage for the factor; these estimates may further be improved if the matrix is sparse. Hence, if $r_{\max} \ll N$, they compare favorably to the (exact) Cholesky factorization, which requires $\mathcal{O}(N^3)$ operations and $\mathcal{O}(N^2)$ storage.

Now, assuming the same block partition (2.1) of A, these algorithms mainly differ by the choice of block rows effectively modified by the $\textsc{approx}_k(\cdot)$ operation. This is motivated by the data-sparse structure of the resulting factors: sequentially semi-separable (SSS) in [16] and hierarchically semi-separable (HSS) in [27]. As will be shown later, this further enables different condition number estimates.

To be more specific, we introduce several possible choices for the block row indices $\mathcal{P}_k$ modified by $\textsc{approx}_k(\cdot)$, as inspired by the above algorithms. Considering first the *SSS choice*, it amounts to perform the approximation on the whole $R_{12}^{(k)}$ block at every step k. In this case, we set $\ell = n - 1$ (no approximation when factorizing the last block) and $\mathcal{P}_k = [1, \ell]$. This situation is illustrated on Figure 1. Note that the factorization in [16] corresponds to this choice and, moreover, uses an orthogonal dropping scheme which preserves a given set of vectors.

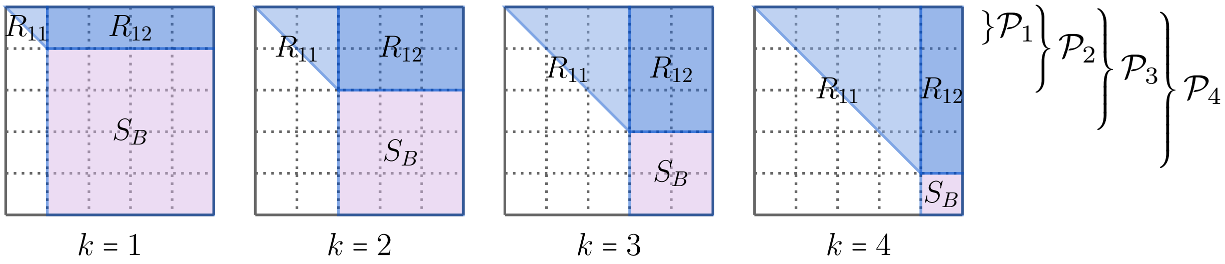

Figure 1: SSS index choice for $n = 5$; note that the R_{12} block is entirely filled with dark blue (dark gray), which means that all rows of the block are approximated.

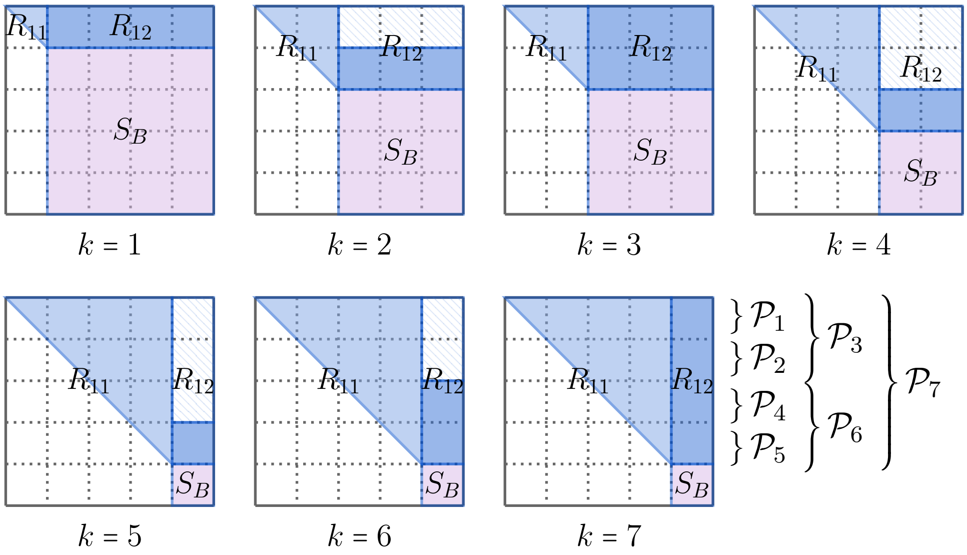

Figure 2: HSS index choice based on a perfect binary tree of height $t = 2$ (the tree is given on the bottom rightmost picture); hatched areas correspond to the rows of R_{12} which are not modified.

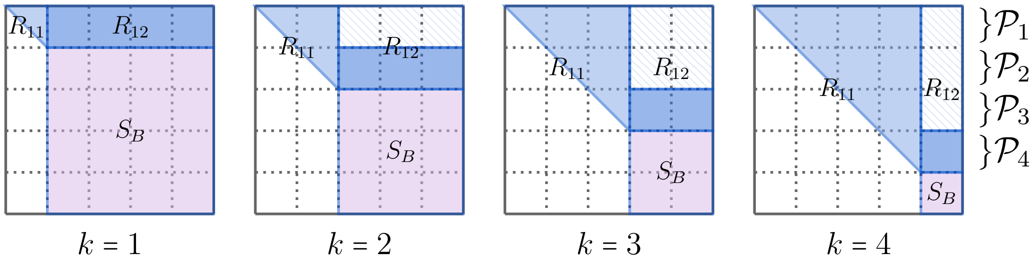

Figure 3: One-level index choice for $n = 5$.

Regarding the *HSS choice*, it requires an auxiliary tree in which every block row $i = 1, ..., n-1$ corresponds to a leaf node. The tree nodes are then ordered in a postorder (that is, children are numbered before their parent; see the bottom right picture of Figure 2 for an example) and each approximation step k corresponds to a tree node with the same number. Note that in this setting ℓ equals to the number of nodes. For every step k, one sets $\mathcal{P}_k = \{i\}$ if k is the tree index of ith leaf, and $\mathcal{P}_k = \cup_{i\in \mathrm{children}(k)}\mathcal{P}_i$ if node k is not a leaf. For simplicity, we only consider perfect binary trees, with hence $n = 2^t + 1$ and $\ell = 2^{t+1} - 1$, where t is the height of the tree. This choice is illustrated on Figure 2 for $t = 2$ with, hence, $n = 5$ and $\ell = 7$. Observe that in this latter case the rows modified during the step $1, 3$ or 4 are not modified during the following step.

Note that the preconditioner in [27] differs from the Algorithm 2.2 with HSS choice for $\mathcal{P}_k$ in that some dropping may also occur in the $R_{11}^{(k)}$ part of the factor. All in all, this additional dropping slightly increase the cost of the factorization (in fact, dropping applies to larger matrices) but is likely to improve the storage requirements (an implementation similar to the one in [27] requires $\mathcal{O}(r_{\max}N\log(N))$ memory if all r_k are the same). The last but not the least, the numerical experiments below indicate that the performance of both methods is similar. We therefore recommend to use the method in [27], while stating that the analysis of HSS choice provide some insight for this method as well.

It should be noted that the complexity estimates mentioned in this subsection require the dimension of blocks in the partition (2.1) of A to be of the order of $r_{\max}$. In a favorable situation when $r_{\max}$ is small, this in turn imply that n is of the same order of magnitude as N, and the number of ℓ approximation steps may become important (see, e.g., Table 1 from Section 4 for an example).

Now, we also consider the case where only newly computed rows are approximated. This amounts to set $\ell = n - 1$ and $\mathcal{P}_k = \{k\}$ for all $1 \le k \le \ell$. Since it corresponds to the HSS choice of $\mathcal{P}_k$ where only the steps k corresponding to leafs are kept, we call it *one-level choice* (see Figure 3 for an example).

Note that a situation when all the entries in $R_{12}^{(k)}$ are dropped at every step; that is, when $\widetilde{R}_{12}^{(k)} = O$ (or, equivalently, $\textsc{approx}_k(\cdot) = O$) for all k, may also be regarded as a one-level choice. In this latter setting, (2.5) further entails $\widetilde{S}_B^{(k)} = A_{n_k+1:n,n_k+1:n}$ and hence $R = \mathrm{blockdiag}(R_{ii})$, where $R_{ii}^T R_{ii} = A_{ii}$. In other words, the resulting preconditioner is given by the block-diagonal part of A as induced by the partitioning (2.1). As a result, the incomplete Cholesky algorithm and the corresponding analysis below also cover these block-diagonal preconditioners.

Eventually, for all the above choices we further set $n_k = \max(\mathcal{P}_k)$, $k = 1, .., \ell$. This means that all the rows computed during the step k (lines 1a-1c of the Algorithm 2.2) will be modified during the subsequent approximation step (line 1d).

3 Analysis

We first recall in the following lemma some basic facts about the Schur complement. In particular, the first statement relates the Schur complement as appearing for instance in (2.3) to its more common form.

Lemma 3.1. *Let A be $N \times N$ and satisfy*

$$A = \begin{pmatrix} R_{11}^T & \\ R_{12}^T & S_A \end{pmatrix} \begin{pmatrix} R_{11} & R_{12} \\ & I \end{pmatrix} \tag{3.1}$$

for some R_{11}, R_{12} and S_A of order $M \times M$, $M \times (N-M)$ and $(N-M) \times (N-M)$, respectively.

(a) If

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{12}^T & A_{22} \end{pmatrix} \tag{3.2}$$

is the 2×2 block partition induced by the partition in (3.1) and if R_{11} is invertible, then

$$S_A = A_{22} - A_{12}^T A_{11}^{-1} A_{12} \, .$$

(b) A is SPD if and only if both $R_{11}^T R_{11}$ and S_A are SPD.

(c) If R_{11} and S_A are invertible, then

$$A^{-1} = \begin{pmatrix} * & * \\ * & S_A^{-1} \end{pmatrix} .$$

Proof. The first statement follows by direct computation, noting that $A_{11} = R_{11}^T R_{11}$ is invertible if R_{11} is. To proof the second we first note that, as stems from (3.1), A is symmetric if and only if S_A is. The proof then follows from [1, Corrolary 3.8]. The last statement is obtained from equation (3.4) in the same reference. ∎

Now, we make use of the following relation between the Schur complements $S_B^{(k)}$, $\widetilde{S}_B^{(k)}$ before and after the approximation step, and the corresponding approximation error $R_{12}^{(k)} - \widetilde{R}_{12}^{(k)}$:

$$\widetilde{S}_B^{(k)} = S_B^{(k)} + \left(R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} \right)^T \left(R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} \right) . \tag{3.3}$$

It follows directly from the definitions (2.7), (2.5) of $S_B^{(k)}$ and $\widetilde{S}_B^{(k)}$ together with the orthogonality condition (2.9). As an important consequence, we note that all B_k , $k = 1,...,\ell$, are SPD and the Algorithm 2.2 is breakdown free; that is, it always produces a preconditioner in the factored form. This property was already observed in [16, 27] for the related variants, and we briefly recall it in Lemma 3.2 below, together with some auxiliary results.

Lemma 3.2. *Let A be SPD and partitioned as in* (2.1)*. Let B_k, $\widetilde{S}_B^{(k)}$ and $S_B^{(k)}$, $k = 1,...,\ell$, be defined by* (2.4), (2.5) *and* (2.7)*, respectively, where $R_{11}^{(k)}$ and $\widetilde{R}_{12}^{(k)}$ stand for, respectively, $R_{1:n_k,1:n_k}$ and $R_{1:n_k,n_k+1:n}$, as defined at the end of step k of Algorithm 2.2 applied to A, and where $R_{12}^{(k)}$ stands for $R_{1:n_k,n_k+1:n}$ as defined prior to the line 1d of the same step. Let* (3.3) *hold for all $1 \le k \le \ell$.*

Then

(a) B_k, $k = 1,...,\ell$, is SPD.

(b) Algorithm 2.2 does not breakdown.

(c) There holds

$$\text{blockdiag}(B_k) = (\ A_{1,1}\ ,\ \cdots\ ,\ A_{n_k,n_k}\ ,\ A_{n_k+1:n,n_k+1:n}\ )\ , \quad k = 1,...,\ell\,. \tag{3.4}$$

Proof. To prove the first statement, we note that, if B_{k-1} is SPD, then so are ${R_{11}^{(k)}}^T R_{11}^{(k)}$ and the Schur complement $S_B^{(k)}$, as follows from Lemma 3.1(b) applied to (2.6). It further follows from (3.3) that $\widetilde{S}_B^{(k)}$ is SPD, and applying again Lemma 3.1(b) to (2.4) shows that B_k is also SPD. Now, the statement (a) follows from this recursive argument and the fact that $B_0 = A$.

Regarding the second statement, we note that Algorithm 2.2 may only break down at line 1b, failing to find R_S such that $R_S^T R_S = R_{n_k,n_k}$. However, one may check that R_{n_k,n_k} is the leading block of the Schur complement $\widetilde{S}_B^{(k-1)}$, and, since the latter is SPD, its leading block may always be factorized.

Eventually, we prove (3.4). Let $B_k = ((B_k)_{i,j})$ be the partitioning into $n \times n$ blocks induced by (2.1). Then, there holds

$$B_k = \begin{pmatrix} (B_{k-1})_{1:n_k,1:n_k} & * \\ * & A_{n_k+1:n,n_k+1:n} \end{pmatrix},$$

where the top left block follows from the comparison of (2.4) and (2.6), whereas the bottom right stems from (2.4) together with (2.5). Applying the above result for $k = 1,...,\ell$ and using $B_0 = A$ finish the proof of (3.4). ∎

We are now ready to prove our main theorem. It assumes that the first n_k block rows of the factor, computed during the first k steps, are not modified over the following p steps (this is always true if $p = 0$). The extreme eigenvalues of $B_{k+p}^{-1} B_{k-1}$

are then related to those of $B_{k+p}^{-1}B_k$ with the help of an additional parameter γ_k as defined in (3.7). Note that this latter involves the Schur complement of B_k and the truncation error of the kth approximation stage, and in some sense measures the approximation accuracy.

Theorem 3.3 (main theorem). *Let A be SPD and let B_k and $\widetilde{S}_B^{(k)}$, $k = 1,\ldots,\ell$, be defined by* (2.4) *and* (2.5)*, respectively, where $R_{11}^{(k)}$ and $\widetilde{R}_{12}^{(k)}$ stand for, respectively, $R_{1:n_k,1:n_k}$ and $R_{1:n_k,n_k+1:n}$, as defined at the end of step k of Algorithm 2.2 applied to A; let $R_{12}^{(k)}$ be given by $R_{1:n_k,n_k+1:n}$ as defined prior to the line 1d of the same step. For some $k > 0$ and $p \geq 0$ such that $k+p \leq \ell$ let* $\textsc{approx}_s(\cdot)$, *$k+1 \leq s \leq k+p$, satisfy the orthogonality condition* (2.9) *and only modify the block rows of $R_{12}^{(s)}$ with indices below n_k.*

Then, setting $\lambda_{\max}^{(k,k+p)} = \lambda_{\max}\left(B_{k+p}^{-1}B_k\right)$ and $\lambda_{\max}^{(k-1,k+p)} = \lambda_{\max}\left(B_{k+p}^{-1}B_{k-1}\right)$ and using similar notation for the minimal eigenvalues, there holds

$$\lambda_{\max}^{(k,k+p)} \leq \lambda_{\max}^{(k-1,k+p)} \leq \lambda_{\max}^{(k,k+p)} + g(\lambda_{\max}^{(k,k+p)}, \gamma_k), \tag{3.5}$$

$$\lambda_{\min}^{(k,k+p)} \geq \lambda_{\min}^{(k-1,k+p)} \geq \lambda_{\min}^{(k,k+p)} - g(\lambda_{\min}^{(k,k+p)}, \gamma_k), \tag{3.6}$$

where

$$\gamma_k = \left\| \left(R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} \right) \widetilde{S}_B^{(k)\,-1/2} \right\| < 1, \tag{3.7}$$

and

$$g(\lambda,\gamma) = \max_{\beta>0} \frac{2\gamma\beta - |\lambda-1|\beta^2}{\beta^2 + \lambda^{-1}}. \tag{3.8}$$

Moreover, if $p = 0$, right inequalities (3.5), (3.6) *become equalities.*

Proof. We only prove inequalities in (3.5), the proof of (3.6) follows the same lines. Considering first right inequality (3.5), we recall that B_{k-1}, B_k which satisfy the assumptions of the theorem also satisfy (2.6), (2.4), which amount to

$$B_{k-1} = \begin{pmatrix} R_{11}^{(k)\,T} & \\ R_{12}^{(k)\,T} & S_B^{(k)} \end{pmatrix} \begin{pmatrix} R_{11}^{(k)} & R_{12}^{(k)} \\ & I \end{pmatrix}, \quad B_k = \begin{pmatrix} R_{11}^{(k)\,T} & \\ \widetilde{R}_{12}^{(k)\,T} & \widetilde{S}_B^{(k)} \end{pmatrix} \begin{pmatrix} R_{11}^{(k)} & \widetilde{R}_{12}^{(k)} \\ & I \end{pmatrix},$$

where $R_{11}^{(k)}$ is $n_k \times n_k$, $R_{12}^{(k)}$ is $n_k \times (n-n_k)$ and $S_B^{(k)}$, $\widetilde{S}_B^{(k)}$ are $(n-n_k)\times(n-n_k)$, all dimensions being considered blockwise. Since the first n_k block rows are not modified during the following p steps, we also have

$$B_{k+p} = \begin{pmatrix} R_{11}^{(k)\,T} & \\ \widetilde{R}_{12}^{(k)\,T} & \overline{S} \end{pmatrix} \begin{pmatrix} R_{11}^{(k)} & \widetilde{R}_{12}^{(k)} \\ & I \end{pmatrix}$$

for some $\overline{S}$ of order $(n-n_k)\times(n-n_k)$. Note that either $\overline{S}=\widetilde{S}_B^{(k)}$ or both matrices have the same 1×1 leading block, which is not modified after the step k of the algorithm. In either case, one further has

$$\lambda_{\max}(\overline{S}^{-1}\,\widetilde{S}_B^{(k)}) \;\geq\; 1\,. \tag{3.9}$$

Now, letting

$$J \;=\; \begin{pmatrix} R_{11}^{(k)\;-1} & -\,R_{11}^{(k)\;-1}\,\widetilde{R}_{12}^{(k)} \\ & I \end{pmatrix},$$

the next equalities follow by direct computation

$$\begin{aligned} J^T B_k J &= \operatorname{diag}(\,I\,,\;\widetilde{S}_B^{(k)}\;)\,, \\ J^T B_{k+p} J &= \operatorname{diag}(\,I\,,\,\overline{S}\,)\,. \end{aligned} \tag{3.10}$$

Using (3.9), this further entails

$$\lambda_{\max}^{(k,\,k+p)} = \lambda_{\max}(B_{k+p}^{-1}B_k) = \lambda_{\max}\left(\,(J^T B_{k+p}^{-1}J)^{-1}J^T B_k J\,\right) = \lambda_{\max}(\overline{S}^{-1}\widetilde{S}_B^{(k)})\,. \tag{3.11}$$

On the other hand, direct computation together with the use of (3.3) (following itself from (2.9)) for the bottom right block leads to

$$J^T B_{k-1} J \;=\; \begin{pmatrix} I & R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} \\ R_{12}^{(k)\,T} - \widetilde{R}_{12}^{(k)\,T} & \widetilde{S}_B^{(k)} \end{pmatrix}. \tag{3.12}$$

Hence, using this latter together with (3.10) and

$$\mathbf{v}_1^T\left(R_{12}^{(k)} - \widetilde{R}_{12}^{(k)}\right)\mathbf{v}_2 \;\leq\; \gamma_k\|\mathbf{v}_1\|\sqrt{\mathbf{v}_2^T\,\widetilde{S}_B^{(k)}\,\mathbf{v}_2}\,, \tag{3.13}$$

which follow from the definition (3.7) of γ_k, there holds

$$\begin{aligned} \lambda_{\max}^{(k-1,\,k+p)} &= \max_{\mathbf{v}}\frac{\mathbf{v}^T J B_{k-1} J^T\mathbf{v}}{\mathbf{v}^T J B_{k+p} J^T\mathbf{v}} \\ &= \max_{\mathbf{v}_1\,,\,\mathbf{v}_2}\frac{\mathbf{v}_1^T\mathbf{v}_1 + \mathbf{v}_2^T\widetilde{S}_B^{(k)}\mathbf{v}_2 + 2\mathbf{v}_1^T\left(R_{12}^{(k)} - \widetilde{R}_{12}^{(k)}\right)\mathbf{v}_2}{\mathbf{v}_1^T\mathbf{v}_1 + \mathbf{v}_2^T\overline{S}\mathbf{v}_2} &&(3.14) \\ &\leq \max_{\beta>0}\frac{\beta^2 + 1 + 2\gamma_k\beta}{\beta^2 + \lambda_{\max}^{-1}\left(\overline{S}^{-1}\widetilde{S}_B^{(k)}\right)}\,. &&(3.15) \end{aligned}$$

where we have set $\beta^2 = \mathbf{v}_1^T\mathbf{v}_1\left(\mathbf{v}_2^T\widetilde{S}_B^{(k)}\mathbf{v}_2\right)^{-1}$. Right inequality (3.5) then follows from (3.15), (3.11) and (3.9), combined with

$$\lambda + g(\lambda,\gamma) \;=\; \max_{\beta>0}\frac{\beta^2 + 1 + 2\gamma\beta}{\beta^2 + \lambda^{-1}}$$

for any $\lambda \geq 1$.

Now, left inequality (3.5) stems from (3.14) and (3.11) by setting $\mathbf{v}_1 = \mathbf{0}$. On the other hand, note that $J^T B_{k-1} J$ is SPD, and, therefore, the inequality (3.13) holds for some $\gamma_k < 1$.

Eventually, setting $p = 0$ we note that $\overline{S} = \widetilde{S}_B^{(k)}$, and, hence, the vectors $\mathbf{v}_1$, $\mathbf{v}_2$ that lead to an equality in (3.13) also allow to reach an equality between (3.15) and (3.14). This is however the only approximation committed in the proof of right inequality (3.5), which therefore becomes an equality. ■

The next corollary provides with (3.16) a general condition number estimate based solely on the orthogonality assumption (2.9). It corresponds to a repeated application of the above theorem in the case where $p = 0$. Note that since no additional assumption is made on the indices of the modified rows, the result may be applied to all of the index choices described in Section 2.3; however, it suits the best the description of the SSS index choice, as this latter do not intent to leave some rows untouched. Now, this result may also be viewed as an extension of the Proposition 2.1 in [27] to $\ell > 1$.

The corollary also introduces the parameter $\overline{\gamma\ell} = \sum_{k=1}^{\ell} \gamma_k$ which, if bounded away from 1, implies a bounded condition number (see (3.17)). Note that, in most cases, the above condition may be relaxed by requiring that $\overline{\gamma\ell}$ is bounded away from a small integer c. This is possible, for instance, if one may subdivide the interval $[0, \ell]$ into few, say c, contiguous subintervals $[k_i, k_{i+1}]$, $i = 1, ..., c-1$, such that the corresponding $\sum_{k_i+1}^{k_{i+1}} \gamma_k$ are all bounded away from 1. Then, from

$$\kappa(R^{-T} A R^{-1}) = \kappa(B_\ell^{-1} A) \leq \kappa(B_\ell^{-1} B_{k_{c-1}}) \cdots \kappa(B_{k_1}^{-1} A)$$

and the observation that each term may be bounded similarly to (3.17) it follows that the overall condition number remains bounded above.

Corollary 3.4 (SSS bound). *Let R be an upper triangular matrix obtained by applying Algorithm 2.2 to an SPD matrix A, with* APPROX$_k(\cdot)$, *$k = 1, \ldots, \ell$, satisfying the orthogonality condition* (2.9)*. Let γ_k, $k = 1, \ldots, \ell$, be given by* (3.7)*, with $\widetilde{S}_B^{(k)}$ defined by* (2.5)*, and with $R_{12}^{(k)}$, $\widetilde{R}_{12}^{(k)}$ standing for $R_{1:n_k, n_k+1:n}$ as given, respectively, prior to the line 1d of step k of Algorithm 2.2, and at the end of this step.*

Then

$$\kappa(R^{-T} A R^{-1}) \leq \prod_{k=1}^{\ell} \frac{1 + \gamma_k}{1 - \gamma_k}, \tag{3.16}$$

If $\ell = 1$, inequality (3.16) *becomes an equality.*

Moreover, if $\overline{\gamma\ell} := \sum_{k=1}^{\ell} \gamma_k < 1$, then

$$\kappa(R^{-T} A R^{-1}) \leq \frac{e^{\overline{\gamma\ell}}}{1 - \overline{\gamma\ell}}. \tag{3.17}$$

Proof. We first prove inequality (3.16). For this, note that setting $p = 0$ in Theorem 3.3 entails $\lambda_{\max}^{(k,k+p)} = \lambda_{\min}^{(k,k+p)} = 1$, and, since $g(1, \gamma_k) = \gamma_k$, it follows from (3.5), (3.6) that

$$\begin{aligned} \lambda_{\max}\left(B_k^{-1}B_{k-1}\right) &= 1 + \gamma_k\,, & (3.18)\\ \lambda_{\min}\left(B_k^{-1}B_{k-1}\right) &= 1 - \gamma_k\,, & (3.19) \end{aligned}$$

the equalities stemming from the last statement of the theorem. The repeated application of the above result further leads to

$$\begin{aligned} \lambda_{\max}(R^{-T}AR^{-1}) &= \lambda_{\max}(B_\ell^{-1}B_0) \le \textstyle\prod_{k=1}^{\ell} \lambda_{\max}\left(B_k^{-1}B_{k-1}\right) = \prod_{k=1}^{\ell}(1+\gamma_k)\\ \lambda_{\min}(R^{-T}AR^{-1}) &= \lambda_{\min}(B_\ell^{-1}B_0) \ge \textstyle\prod_{k=1}^{\ell} \lambda_{\min}\left(B_k^{-1}B_{k-1}\right) = \prod_{k=1}^{\ell}(1-\gamma_k) \end{aligned}$$

and the inequality (3.16) readily follows.

Eventually, observing that $1 - \gamma_k \le e^{\gamma_k}$ and $(1-\gamma_k)(1-\gamma_s) \ge 1 - \gamma_k - \gamma_s$ for all $\gamma_k, \gamma_s > 0$, the estimate (3.17) follows directly from (3.16) . ■

Before we consider in more details the case where $p > 0$; that is, the case where some block rows of the factor are not modified for several consecutive steps, we sheds some light in Lemma 3.5 below on how the function $g(\lambda, \gamma)$ from our main theorem depends on λ . Considering first $\lambda \approx 1$, it shows that $g(\lambda, \gamma) \approx \gamma\lambda$ (as may be concluded from the second term in the right hand sides of (3.20), (3.21)). The estimates (3.5), (3.6) then amount to

$$\begin{aligned} \lambda_{\max}^{(k-1,k+p)} &\le (1+\gamma_k)\lambda_{\max}^{(k,k+p)}\,,\\ \lambda_{\min}^{(k-1,k+p)} &\ge (1-\gamma_k)\lambda_{\min}^{(k,k+p)}\,, \end{aligned}$$

where the notation for $\lambda_{\max}^{(k,k+p)} = \lambda_{\max}\left(B_{k+p}^{-1}B_k\right)$, $\lambda_{\min}^{(k,k+p)} = \lambda_{\min}\left(B_{k+p}^{-1}B_k\right)$ is the same as in Theorem 3.3. The contribution of each truncation accuracy γ_k to the bound on the condition number is then essentially the same as obtained by setting $p = 0$ (see Corollary 3.4).

On the other hand, if λ is either large ($\lambda_{\max}^{(k,k+p)}$ case) or small ($\lambda_{\min}^{(k,k+p)}$ case), we observe that $g(\lambda, \gamma) \approx \gamma^2\lambda/|\lambda - 1|$ (this follows from the first term in the right hand sides of (3.20), (3.21)). In particular, if $\lambda_{\min}^{(k,k+p)}$ is small, the lower bound (3.6) on $\lambda_{\min}^{(k-1,k+p)}$ is essentially given by $(1-\gamma_k^2)\lambda_{\min}^{(k,k+p)}$ which compares favorably to the estimate $(1-\gamma_k)\lambda_{\min}^{(k,k+p)}$ from the above inequalities. The improvement is even more important when $\lambda_{\max}^{(k,k+p)}$ is large, since the upper bound (3.5) on $\lambda_{\min}^{(k-1,k+p)}$ then essentially corresponds to $\lambda_{\min}^{(k,k+p)} + \gamma_k^2$ instead of $(1+\gamma_k)\lambda_{\min}^{(k,k+p)}$.

Lemma 3.5. *Let λ and $\gamma \le 1$ be real and positive, and $g(\lambda,\gamma)$ be defined by* (3.8). *Then, $g(1,\gamma) = \gamma$ and, for $\lambda \neq 1$,*

$$g(\lambda,\gamma) \;\le\; \min\left(\gamma^2\frac{\lambda}{|\lambda-1|}\,,\,\gamma\lambda\right), \tag{3.20}$$

$$g(\lambda,\gamma) \;\ge\; \max\left(c_1(\lambda,\gamma)\cdot\gamma^2\frac{\lambda}{|\lambda-1|}\,,\,c_2(\lambda,\gamma)\cdot\gamma\lambda\right), \tag{3.21}$$

where $c_1(\lambda,\gamma) \;=\; |\lambda-1|^2(|\lambda-1|^2+\gamma^2\lambda)^{-1} \to 1$ for either $\lambda \to 0$ or $\lambda \to \infty$ and $c_2(\lambda,\gamma) \;=\; (2-|\lambda-1|/\gamma)(2+\lambda-1)^{-1} \to 1$ for $\lambda \to 1$.

Proof. The proof of $g(1,\gamma) = \gamma$ is straightforward. Now, setting $\widetilde{\beta} = \gamma|\lambda-1|^{-1}$, the first term in the maximum (3.21) follows from

$$g(\lambda,\gamma) \;=\; \max_{\beta>0}\frac{2\gamma\beta-|\lambda-1|\beta^2}{\beta^2+\lambda^{-1}} \;\ge\; \frac{2\gamma\widetilde{\beta}-|\lambda-1|\widetilde{\beta}^2}{\widetilde{\beta}^2+\lambda^{-1}} \;=\; \frac{\gamma^2\lambda|\lambda-1|}{|\lambda-1|^2+\gamma^2\lambda}\,.$$

Using the same reasoning with $\widetilde{\beta} = 1$ instead leads to the second term.

To prove (3.20), note that $2\gamma\beta - |\lambda-1|\beta^2 \;\le\; \gamma^2|\lambda-1|^{-1}$ holds for all β, and the first term in the minimum (3.20) follows from

$$g(\lambda,\gamma) \;=\; \max_{\beta>0}\frac{2\gamma\beta-|\lambda-1|\beta^2}{\beta^2+\lambda^{-1}} \;\le\; \gamma^2\max_{\beta>0}\frac{|\lambda-1|^{-1}}{\beta^2+\lambda^{-1}} \;=\; \gamma^2\frac{\lambda}{|\lambda-1|}\,.$$

Noting that $|\lambda-1| \ge \gamma(1-\lambda)$, the second term follows from

$$\begin{aligned} g(\lambda,\gamma) \;=\; \max_{\beta>0}\frac{2\gamma\beta-|\lambda-1|\beta^2}{\beta^2+\lambda^{-1}} \;&\le\; \max_{\beta>0}\frac{2\gamma\beta-\gamma(1-\lambda)\beta^2}{\beta^2+\lambda^{-1}} \\ &=\; \max_{\beta>0}\frac{-\gamma(\beta-1)^2+\gamma\lambda(\beta^2+\lambda^{-1})}{\beta^2+\lambda^{-1}} \;\le\; \gamma\lambda\,. \qquad \blacksquare \end{aligned}$$

Now, we make the above discussion more specific by assuming that $\ell = n-1$, $n_k = k$ for all $k = 1,\ldots,\ell$, and further, that only the block row n_k; that is, only the block row of the factor computed during the step k, is modified during this step. Corollary 3.6 then shows (see (3.22)) that the condition number is bounded (up to a "penalization" factor $2+\widetilde{\gamma}^2\ell$) by a quotient of two quantities, one being essentially a sum of γ_k^2, the other roughly corresponding (if $c_* \approx 1$) to a product of $1-\gamma_k^2$ (with, however, $1-\gamma_{\max}^2$ instead of $1-\gamma_1^2$). Note that, although this estimate is asymptotically better, it may be less accurate than (3.16) for small ℓ; in particular, it is always less accurate for $\ell = 1$.

Next, the inequality (3.23) further highlights the role played by the parameter $\widetilde{\gamma}^2\ell = \sum_{k=1}^{\ell}\gamma_k^2$: if this latter is bounded away from 1, then the condition number is also bounded. Note that this condition is less restrictive that the one required by Corollary 3.4, namely that $\overline{\gamma}\ell = \sum_{k=1}^{\ell}\gamma_k$ should be away from 1; this comes with the additional assumption on the indices of the modified rows. On the other hand, as in Corollary 3.4, the requirement of $\widetilde{\gamma}^2\ell$ being bounded away from one may often be relaxed by using essentially the same arguments as stated there.

Eventually, we note that the corollary below may only be applied to the one-level index choice. Whereas it provides some insight on how the condition number behaves in this case, we still advocate the direct use of eigenvalue bounds (3.5), (3.6) if one needs to obtain an accurate estimate of the condition number; in the one-level case, this amounts to compute, for instance, $\widetilde{\lambda}_{\max}^{(\ell-1,\ell)}$ using $\widetilde{\lambda}_{\max}^{(\ell,\ell)} = 1$ and γ_ℓ, then compute $\widetilde{\lambda}_{\max}^{(\ell-2,\ell)}$ using $\widetilde{\lambda}_{\max}^{(\ell-1,\ell)}$ and $\gamma_{\ell-1}$, and so on, until $\lambda_{\max}(R^{-T}AR^{-1}) = \widetilde{\lambda}_{\max}^{(0,\ell)}$ is obtained. This procedure is used, in particular, for the numerical experiments in Section 4 . Note that the use of tilde emphasizes that $\widetilde{\lambda}$ is just a bound on the actual eigenvalue λ .

Corollary 3.6 (one-level bound)**.** *Let the assumption of Corollary 3.4 hold. In addition, let $\ell = n-1$, $n_k = k$ and* $\text{APPROX}_k(\cdot)$*, $k = 1,..,\ell$, only modify the block row n_k.*

Then, setting $\widetilde{\gamma}^2\ell := \sum_{k=1}^{\ell}\gamma_k^2$ and $\gamma_{\max} = \max_{1\le k\le\ell}\gamma_k$, there holds

$$\kappa(R^{-T}AR^{-1}) \;\le\; (2+\widetilde{\gamma}^2\ell)\cdot\frac{\left(1+\sqrt{\widetilde{\gamma}^2\ell}\right)^2}{(1-\gamma_{\max}^2)\prod_{k=2}^{\ell}\left(1-\frac{\gamma_k^2}{c_*}\right)}\,, \tag{3.22}$$

where $c_* = (1+\widetilde{\gamma}^2\ell)/(1+\widetilde{\gamma}^2\ell+1-\gamma_{\max}^2) \to 1$ when $\widetilde{\gamma}^2\ell \to \infty$.

Moreover, if $\widetilde{\gamma}^2\ell < 1$, then

$$\kappa(R^{-T}AR^{-1}) \;\le\; \left(\frac{1+\sqrt{\widetilde{\gamma}^2\ell}}{1-\sqrt{\widetilde{\gamma}^2\ell}}\right)^2 . \tag{3.23}$$

Proof. We first show that the results follow from

$$\lambda_{\max}(R^{-T}AR^{-1}) \;=\; \lambda_{\max}^{(0,\ell)} \;<\; \left(1+\sqrt{\widetilde{\gamma}^2\ell}\right)^2, \tag{3.24}$$

$$\lambda_{\min}(R^{-T}AR^{-1}) \;=\; \lambda_{\min}^{(0,\ell)} \;\ge\; \max_{1\,>\,c\,\ge\,\gamma_{\max}}(1-c)\prod_{k=2}^{\ell}\left(1-\frac{\gamma_k^2}{c}\right), \tag{3.25}$$

proving these inequalities later. The estimate (3.22) follows from (3.24), (3.25) setting $c = c_*$ and using

$$1-c_* \ge \frac{1-\gamma_{\max}^2}{1+\widetilde{\gamma}^2\ell+1-\gamma_{\max}^2} > \frac{1-\gamma_{\max}^2}{2+\widetilde{\gamma}^2\ell} > 0\,.$$

This latter further shows that $c_* < 1$, whereas $c_* \geq \gamma_{\max}$ follows from

$$c_* - \gamma_{\max} = (1 - \gamma_{\max}) \frac{1 - \gamma_{\max} + \widetilde{\gamma}^2 - \gamma_{\max}^2 \ell}{2 + \widetilde{\gamma}^2 \ell - \gamma_{\max}^2} > 0 \, .$$

On the other hand, estimate (3.23) is obtained setting $c_+ = (\widetilde{\gamma}^2 \ell)^{1/2} > \gamma_{\max}$. Note that $c_+ < 1$ by assumption, whereas (3.25) further implies

$$\lambda_{\min}(R^{-T} A R^{-1}) \geq (1 - c_+) \prod_{k=1}^{\ell} \left(1 - \frac{\gamma_k^2}{c_+}\right) \geq (1 - c_+)\left(1 - \sum_{k=1}^{\ell} \frac{\gamma_k^2}{c_+}\right) = \left(1 - \sqrt{\widetilde{\gamma}^2 \ell}\right)^2 \, .$$

Now, note that the assumptions on $\textsc{approx}_k(\cdot)$ made here satisfy the requirements of Theorem 3.3 for all possible values of k and p; that is, for all $1 \leq k \leq \ell$ and $0 \leq p \leq \ell - k$.

We begin with the prove of (3.24). First observe that the upper bound in (3.5) is an increasing function of $\lambda_{\max}^{(k,\,k+p)}$. Hence, setting $\widetilde{\lambda}_{\max}^{(\ell,\,\ell)} = b > 1$ (instead of 1) and applying the recursion (3.5) to define the remaining $\widetilde{\lambda}_{\max}^{(k-1,\,\ell)}$,$k = 1, ..., \ell$, as follows

$$\widetilde{\lambda}_{\max}^{(k-1,\,\ell)} = \widetilde{\lambda}_{\max}^{(k,\,\ell)} + g(\widetilde{\lambda}_{\max}^{(k,\,\ell)}, \gamma_k) \, , \tag{3.26}$$

one concludes that $\widetilde{\lambda}_{\max}^{(0,\ell)} > \lambda_{\max}^{(0,\ell)}$. On the other hand, (3.26) also entails

$$\widetilde{\lambda}_{\max}^{(0,\ell)} \geq \cdots \geq \widetilde{\lambda}_{\max}^{(\ell,\ell)} = b \, ,$$

which, together with (3.26) and $g(\lambda, \gamma) \leq \gamma^2 \lambda |\lambda - 1|^{-1}$ (as follows from (3.20)) implies

$$\widetilde{\lambda}_{\max}^{(k-1,\ell)} \leq \widetilde{\lambda}_{\max}^{(k,\ell)} + \gamma_k^2 \frac{b}{b-1} \, ,$$

and hence

$$\lambda_{\max}^{(0,\ell)} \leq \widetilde{\lambda}_{\max}^{(0,\ell)} \leq b + \frac{b}{b-1} \textstyle\sum_{k=1}^{\ell} \gamma_k^2 = b + \frac{b}{b-1} \widetilde{\gamma}^2 \ell \, . \tag{3.27}$$

Setting $b = 1 + (\widetilde{\gamma}^2 \ell)^{1/2}$ finishes the proof of (3.24). One may further check that this choice of b maximizes the bound in (3.27).

The proof for the lower bound (3.25) follows similar lines. First, set $\widetilde{\lambda}_{\max}^{(\ell-1,\ell)} = 1 - c$, and observe that $\widetilde{\lambda}_{\max}^{(\ell-1,\ell)} \geq \lambda_{\max}^{(\ell-1,\ell)}$ since $c \geq \max_{k=1,\dots,\ell} \gamma_k$. Further, define

$$\widetilde{\lambda}_{\min}^{(k-1,\ell)} = \widetilde{\lambda}_{\min}^{(k,\ell)} - g(\widetilde{\lambda}_{\min}^{(k,\ell)}, \gamma_k) \, ,$$

with, hence, $\lambda_{\max}^{(0,\ell)} \geq \widetilde{\lambda}_{\max}^{(0,\ell)}$ and $\widetilde{\lambda}_{\min}^{(k,\ell)} \leq 1 - c$, $k = 1, \dots, \ell - 1$. Using this latter together with $g(\lambda, \gamma) \leq \gamma^2 \lambda |\lambda - 1|^{-1}$ (see (3.20)) entails

$$\widetilde{\lambda}_{\min}^{(k-1,\ell)} \geq \widetilde{\lambda}_{\min}^{(k,\ell)} \left(1 - \frac{\gamma_k^2}{c}\right) \, ,$$

and hence

$$\lambda_{\max}^{(0,\ell)} \geq \widetilde{\lambda}_{\max}^{(0,\ell)} \geq (1-c)\prod_{k=2}^{\ell}\Big(1-\frac{\gamma_k^2}{c}\Big)\,, \tag{3.28}$$

and the inequality (3.25) follows. ■

Now, regarding the *HSS* choice, we note that the derivation of an analytical bound similar to (3.22) seems more involved and would perhaps provide less insight. However, by analogy to the one-level choice, such a bound may be computed numerically, using again the estimates from Theorem 3.3. Since a proper use of these latter (which allows a better estimate than that given by (3.16)) is less straightforward in the HSS case, we outline the main ideas below. To begin with, observe that the indices $\mathcal{P}_k$ of block rows approximated during the step k of Algorithm 2.2 (and associated to the node k in the corresponding HSS tree, see, e.g., Figure 2) are modified again only during the parent step; that is, during the step k_p associated to the parent of the node k in the tree. Hence, the estimates (3.5), (3.6) may be applied for this k with $p \leq k_p - k - 1$. This allows to use the recursive procedure summarized in Algorithm 3.1, which yields an upper bound $\widetilde{\lambda}_{\max}^{(k-1,\,k+s)}$ on $\lambda_{\max}^{(k-1,\,k+s)}$ for any $k \geq 1$ and $s \geq -1$ (the algorithm for the lower bound $\widetilde{\lambda}_{\min}^{(k-1,\,k+s)}$ is similar, with line 5 based on (3.6) instead of (3.5)). Note that the procedure implicitly relies on the fact that $\lambda + g(\lambda,\gamma)$ is an increasing function of $\lambda \geq 1$.

Algorithm 3.1 (HSS Bound on $\lambda_{\max}^{(k-1,\,k+s)}$) : $\widetilde{\lambda}_{\max}^{(k-1,\,k+s)} = \text{COND}\,(k-1\,,\,k+s)$

```
1.  if ($s = -1$) : return 1
2.  if ($s = 0$) : return $1 + \gamma_k$
3.  $p = \min(\ \text{parent}\,(k) - k - 1\ ,\ s\ )$
4.  $\widetilde{\lambda}_{\max}^{(k,\,k+p)} = \text{COND}\,(k\,,\,k+p)$
5.  $\widetilde{\lambda}_{\max}^{(k-1,\,k+p)} = \widetilde{\lambda}_{\max}^{(k,\,k+p)} + g(\widetilde{\lambda}_{\max}^{(k,\,k+p)}, \gamma_k)$
6.  $\widetilde{\lambda}_{\max}^{(k+p,\,k+s)} = \text{COND}\,(k+p\,,\,k+s)$
7.  return $\widetilde{\lambda}_{\max}^{(k-1,\,k+p)}\,\widetilde{\lambda}_{\max}^{(k+p,\,k+s)}$
```

Let us now turn to the preconditioner corresponding to the block-diagonal part of A. It has been observed in Section 2.3 that this preconditioner may be obtained with Algorithm 2.2 by setting $\widetilde{R}_{12}^{(k)} = O$ and that it corresponds to the one-level index choice. We now compare the results in this paper with the existing analysis of the block-diagonal preconditioners as summarized in [1, Chapter 9]. Assuming that A has a block partition (2.1), this latter analysis relies at step k on the 2×2

partitioning of $A^{(k)} = A_{k:n,k:n}$ given by

$$A^{(k)} = \begin{pmatrix} A_{k,k} & A_{k,k+1:n} \\ A^T_{k,k+1:n} & A^{(k+1)} \end{pmatrix} .$$

The corresponding block-diagonal preconditioner is then defined by

$$D_k = \begin{pmatrix} A_{k,k} & \\ & A^{(k+1)} \end{pmatrix} ,$$

and it is known that

$$\kappa(D_k^{-1} A^{(k)}) = \frac{1+\gamma_k^{\mathrm{CBS}}}{1-\gamma_k^{\mathrm{CBS}}} , \tag{3.29}$$

where

$$\gamma_k^{\mathrm{CBS}} = \max_{\mathbf{v}_1,\mathbf{v}_2} \frac{\mathbf{v}_1^T A_{k,k+1:n} \mathbf{v}_2}{\{\mathbf{v}_1^T A_{k,k} \mathbf{v}_1 \cdot \mathbf{v}_2^T A^{(k+1)} \mathbf{v}_2\}^{1/2}} ,$$

is the so-called Cauchy–Bunyakovsky–Schwarz (CBS) constant for the partitioning.

On the other hand, combining Lemma 3.2(c) and the fact that $\widetilde{R}_{12}^{(k)} = O$, $k = 1, \dots, \ell$, yields

$$B_k = \operatorname{blockdiag}\left(A_{1,1} , \cdots , A_{k,k} , A^{(k+1)} \right) ,$$

and, hence,

$$\frac{1+\gamma_k}{1-\gamma_k} = \kappa(B_k^{-1} B_{k-1}) = \kappa(D_k^{-1} A^{(k)}) = \frac{1+\gamma_k^{\mathrm{CBS}}}{1-\gamma_k^{\mathrm{CBS}}} ,$$

where the first equality follows from the last statement of Theorem 3.3 and $g(1,\gamma_k) = \gamma_k$. Clearly, this is only possible if $\gamma_k = \gamma_k^{\mathrm{CBS}}$. Note that the repeated use of the 2×2 bound (3.29) leads to the same estimate as (3.16). However, the assymptotically sharper bound (3.22) is also applicable here, and an even sharper estimate may be obtained by repeated application of (3.5), (3.6) with $p = \ell - k$, both requiring the same CBS constants. To the best of our knowledge, this improved bounds are seemingly presented here for the first time.

Note that one can hardly find a better estimate for the one-level case that the one based on (3.5), (3.6), and which would only involve individual accuracy measures γ_k, $k = 1, \dots, \ell$, since this latter is sharp. More precisely, Theorem 3.7 below states that for any set γ_k, $k = 1, \dots, \ell$, of positive values smaller than 1 there is a matrix $A = A(\gamma_1, \cdots, \gamma_\ell)$ for which the incomplete Cholesky factorization algorithm performs approximations with accuracies γ_k and produce a squene $B_1, \dots, B_\ell$ such that right inequalities (3.5), (3.6) simultaneously become equalities for all k. In other words, for any possible bound which may be obtained using (3.5), (3.6) there is a matrix which allows to reach it.

Now, Theorem 3.7 is formulated in the context of block-diagonal preconditioners, that is, it assumes that Algorithm 2.2 is considered with $\text{APPROX}_k(\cdot) = O$. This allows to show that the sharpness property holds for this subclass of one-level preconditioners. The above assumption on the approximation operation is however not restrictive in practice since known approximation procedures, and in particular those based on the orthogonal decomposition as described in Section 2.2, amount to all-zero approximation if the corresponding threshold is chosen small enough.

Theorem 3.7. *For any set of positive values $\widetilde{\gamma}_k < 1$, $k = 1,...,\ell$, there exist an SPD matrix $A = A(\widetilde{\gamma}_1,\cdots,\widetilde{\gamma}_\ell)$ partitioned as in* (2.1) *such that the application of Algorithm 2.2 to $A(\widetilde{\gamma}_1,\cdots,\widetilde{\gamma}_{\ell-1})$ with $\text{APPROX}_k(\cdot) = O$, $k = 1,\dots,\ell$, returns an upper triangular R and there holds:*

(a)
$$\gamma_k \;=\; \widetilde{\gamma}_k\,, \qquad \textit{for } k = 1\,,\dots\,,\ell\,,$$

where γ_k is given by (3.7)*, with $\widetilde{S}_B^{(k)}$ being defined by* (2.5)*, with $\widetilde{R}_{12}^{(k)}$ standing for $R_{1:n_k,n_k+1:n}$ at the end of step k of the algorithm and with $R_{12}^{(k)}$ standing for $R_{1:n_k,n_k+1:n}$ being defined prior to the line 1d of the same step;*

(b) setting $\lambda_{\max}^{(k,\ell)} = \lambda_{\max}(B_\ell^{-1}B_k)$ and $\lambda_{\min}^{(k,\ell)} = \lambda_{\min}(B_\ell^{-1}B_k)$, $k = 1,\dots,\ell$, where B_k is defined by (2.4)*, with $\widetilde{R}_{12}^{(k)}$ defined as in (a) and with $R_{11}^{(k)}$ standing for $R_{1:n_k,1:n_k}$ at the end of step k of the algorithm, there holds*

$$\lambda_{\max}^{(k-1,\ell)} \;=\; \lambda_{\max}^{(k,\ell)} + g(\,\lambda_{\max}^{(k,\ell)}\,,\widetilde{\gamma}_k\,)\,, \tag{3.30}$$

$$\lambda_{\min}^{(k-1,\ell)} \;=\; \lambda_{\min}^{(k,\ell)} - g(\,\lambda_{\min}^{(k,\ell)}\,,\widetilde{\gamma}_k\,)\,. \tag{3.31}$$

Proof. We prove the theorem by induction for the case when A has order $(2\ell{+}2)\times(2\ell{+}2)$ and is partitioned into $(\ell{+}1)\times(\ell{+}1)$ blocks of size 2. Moreover, the block-diagonal part of A is chosen to be identity matrix, with hence $B_\ell = R^T R = I$.

First, the basic case $\ell = 1$ is proved using

$$A(\widetilde{\gamma}_1) \;=\; \begin{pmatrix} I & -\widetilde{\gamma}_1 I \\ -\widetilde{\gamma}_1 I & I \end{pmatrix},$$

since $\gamma_1 = \widetilde{\gamma}_1$ and, by Theorem 3.3, we have $\lambda_{\max}^{(0,1)} = \lambda_{\max}\,(\,A(\widetilde{\gamma}_1)\,) \;=\; 1 + \widetilde{\gamma}_1$ and $\lambda_{\min}^{(0,1)} = \lambda_{\min}\,(\,A(\widetilde{\gamma}_1)\,) \;=\; 1 - \widetilde{\gamma}_1$.

Now, let $A(\widetilde{\gamma}_2,\cdots,\widetilde{\gamma}_\ell)$ be the matrix which satisfy the assumption of the theorem for $\ell - 1$ and the values $\widetilde{\gamma}_2,\dots,\widetilde{\gamma}_\ell$. We prove below that the theorem holds for ℓ

and the values $\widetilde{\gamma}_1, \ldots, \widetilde{\gamma}_\ell$ using a matrix $A(\widetilde{\gamma}_1, \cdots, \widetilde{\gamma}_\ell) = A$ with

$$A \;=\; \begin{pmatrix} 1 & & \widetilde{\gamma}_1 \sigma_{\max}^{1/2} \mathbf{v}_{\max}^T \\ & 1 & \widetilde{\gamma}_1 \sigma_{\min}^{1/2} \mathbf{v}_{\min}^T \\ \widetilde{\gamma}_1 \sigma_{\max}^{1/2} \mathbf{v}_{\max} & \widetilde{\gamma}_1 \sigma_{\min}^{1/2} \mathbf{v}_{\min} & A(\widetilde{\gamma}_2, \cdots, \widetilde{\gamma}_\ell) \end{pmatrix},$$

where $\mathbf{v}_{\max}^T$, $\mathbf{v}_{\min}^T$ are two orthogonal unit norm vectors satisfying $A(\widetilde{\gamma}_2, \cdots, \widetilde{\gamma}_\ell)\mathbf{v}_{\max} = \mathbf{v}_{\max}\sigma_{\min}$ and $A(\widetilde{\gamma}_2, \cdots, \widetilde{\gamma}_\ell)\mathbf{v}_{\min} = \mathbf{v}_{\min}\sigma_{\max}$.

First, we show that $\gamma_1 = \widetilde{\gamma}_1$. Note that

$$R_{12}^{(1)} \;=\; \begin{pmatrix} \widetilde{\gamma}_1 \sigma_{\max}^{1/2} \mathbf{v}_{\max} & \widetilde{\gamma}_1 \sigma_{\min}^{1/2} \mathbf{v}_{\min} \end{pmatrix}^T$$

whereas, since $\textsc{approx}_k(\cdot) = O$, it follows from (2.8) that $\widetilde{R}_{12}^{(1)} = O$ and, further from (2.5) that $\widetilde{S}_B^{(1)} = A(\widetilde{\gamma}_2, \cdots, \widetilde{\gamma}_\ell)$. Hence, using the definition (3.7) of γ_k, we have

$$\gamma_1^2 \;=\; \left\| R_{12}^{(1)}\; \widetilde{S}_B^{(1)\,-1/2} \right\|^2 \;=\; \left\| R_{12}^{(1)}\; A(\widetilde{\gamma}_2, \cdots, \widetilde{\gamma}_\ell)^{-1}\; R_{12}^{(1)\,T} \right\| \;=\; \left\| \operatorname{diag}(\widetilde{\gamma}_1^2\,, \widetilde{\gamma}_1^2) \right\| \;=\; \widetilde{\gamma}_1^2\,.$$

The proof of $\gamma_k = \widetilde{\gamma}_k$, $k = 2, \ldots, \ell$ stems from the induction assumption.

Second, we prove (3.30) for $k = 1$ (the proof for $k = 2, \ldots, \ell$ follows from induction assumption, and the proof of (3.31) is similar). Since $B_0 = A$ and $B_\ell = I$, there holds

$$\lambda_{\max}^{(0,\ell)} \;=\; \max_{\mathbf{w}} \frac{\mathbf{w}^T A \mathbf{w}}{\mathbf{w}^T \mathbf{w}^T} \;=\; \max_{\mathbf{w}_1, \mathbf{w}_2} \frac{\mathbf{w}_1^T \mathbf{w}_1 + 2\mathbf{w}_1^T\; R_{12}^{(1)}\; \mathbf{w}_2 + \mathbf{w}_2^T A(\widetilde{\gamma}_2, \cdots, \widetilde{\gamma}_\ell) \mathbf{w}_2}{\mathbf{w}_1^T \mathbf{w}_1 + \mathbf{w}_2^T \mathbf{w}_2}\,. \qquad (3.32)$$

On the other hand, $B_1 = \operatorname{diag}(I\,,\, A(\widetilde{\gamma}_2, \cdots, \widetilde{\gamma}_\ell))$, and hence $\lambda_{\max}^{(1,\ell)} = \sigma_{max}$. Therefore, using $\mathbf{w}_1 = \beta\begin{pmatrix} 1 & 0 \end{pmatrix}^T$ and $\mathbf{w}_2 = \sigma_{\max}^{-1/2} \mathbf{v}_{\max}$ in (3.32) leads to

$$\lambda_{\max}^{(0,\ell)} \;\geq\; \max_{\beta} \frac{\beta^2 + 2\beta\widetilde{\gamma}_1 + 1}{\beta^2 + \lambda_{\max}^{(1,\ell)\,-1}} \;=\; \lambda_{\max}^{(1,\ell)} + g(\lambda_{\max}^{(1,\ell)}, \widetilde{\gamma}_1)\,.$$

The proof is finished by noting that Theorem 3.3 applies in this setting with $k = 1$, $p = \ell - 1$; hence, using (3.5) the above inequality becomes an equality. ■

On the other hand, the following counterexample shows that the condition number estimate obtained in the one-level case by repeated application of (3.5), (3.6) with $p = \ell - k$, $k = 1, \ldots, \ell$, may not hold assuming only (2.9); that is, the additional assumption on the indices of modified block rows (as made for the one-level case) is really necessary. The counterexample also demonstrates that a system preconditioned by the incomplete Cholesky factorization may in principle be more ill-conditioned than the original system; compare $\kappa(A) = 7$ with $\kappa(R^{-T} A R^{-1}) \approx 8.61$.

Example 3.1. *Let $n_1 = 1$, $n_2 = 2$, $n = 3$ and $\ell = 2$. Set*

$$A = \begin{pmatrix} 1 & .4 & .4 \\ .4 & 1 & -.4 \\ .4 & -.4 & 1 \end{pmatrix}, \quad B_1 = \left(\begin{array}{c|cc} 1 & & \\ \hline & 1 & -.4 \\ & -.4 & 1 \end{array}\right),$$

and $B_2 = R^T R$ with

$$R = \left(\begin{array}{cc|c} 1 & & -.16 \\ & 1 & -.32 \\ \hline & & \sqrt{.872} \end{array}\right).$$

One may check that $\gamma_1^2 = 8/15$ and $\gamma_2^2 = 4/109$ and hence

$$\begin{aligned} \widetilde{\lambda}_{\max} &= 1 + \gamma_2 + g(1+\gamma_2, \gamma_1) \lessapprox 1.9\,, \\ \widetilde{\lambda}_{\min} &= 1 - \gamma_2 - g(1-\gamma_2, \gamma_1) \gtrapprox .24\,, \end{aligned}$$

whereas

$$\kappa(R^{-T} A R^{-1}) = \kappa(B_2^{-1} A) \gtrapprox 8.61 > 7.92 \gtrapprox \frac{\widetilde{\lambda}_{\max}}{\widetilde{\lambda}_{\min}}. \qquad \blacksquare$$

Eventually, we provide some practical upper bounds for the accuracy measure γ_k. One obvious way to obtain such an bound is to split the norm of a product of two factors in (3.7) into a product of two norms:

$$\gamma_k \le \left\| R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} \right\| \left\| \widetilde{S}_B^{(k)\,-1/2} \right\| . \tag{3.33}$$

As noted in Section 2.2, the fist factor may be easily controlled by imposing a given threshold tol_a. However, the impact of the second factor is less obvious, since $\widetilde{S}_B^{(k)}$ depends on all the approximations made in the generic Cholesky algorithm before and during the step k. The next theorem is helpful in this respect, since it relates $\widetilde{S}_B^{(k)}$ (and, hence, its norms) to the corresponding Schur complement $S_A^{(k)}$ of A (and their norms), as well as the bottom right subblock of A.

Theorem 3.8. *Let A be SPD and partitioned as in* (2.1)*. Let $\widetilde{S}_B^{(k)}$ be defined by* (2.5)*, with $\widetilde{R}_{12}^{(k)}$ standing for $R_{1:n_k, n_k+1:n}$ at the end of step k of the Algorithm 2.2 applied to A.*

Then

$$\mathbf{v}^T S_A^{(k)} \mathbf{v} \le \mathbf{v}^T\, \widetilde{S}_B^{(k)}\, \mathbf{v} \le \mathbf{v}^T A_{n_k+1:n, n_k+1:n} \mathbf{v} \quad \forall \mathbf{v}\,, \quad k = 1, \dots, \ell \tag{3.34}$$

where $S_A^{(k)} = A_{n_k+1:n, n_k+1:n} - A_{1:n_k, n_k+1:n}^T A_{1:n_k, 1:n_k}^{-1} A_{1:n_k, n_k+1:n}$. Moreover,

$$\gamma_k \le \left\| R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} \right\| \left\| S_A^{(k)\,-1} \right\|^{1/2} \le \left\| R_{12}^{(k)} - \widetilde{R}_{12}^{(k)} \right\| \left\| A^{-1} \right\|^{1/2} . \tag{3.35}$$

Proof. We prove left inequality (3.34) by induction. For $k = 1$, $\mathbf{v}^T \widetilde{S}_B^{(1)} \mathbf{v} \geq \mathbf{v}^T S_B^{(1)} \mathbf{v} = \mathbf{v}^T S_A^{(1)} \mathbf{v}$, where the first inequality stems from (3.3) and the equality follows since by (2.6) $S_B^{(1)}$ is a Schur complement of $B_0 = A$.

Now, assume that (3.34) holds for a given k. First, note that both $S_B^{(k+1)}$ and $\widetilde{S}_B^{(k)}$ are the Schur complements of B_k, $k = 1, ..., \ell-1$, as follows from (2.6) and (2.4), respectively. Hence, according to Lemma 3.1(c), $S_B^{(k+1)}$ is a Schur complement of $\widetilde{S}_B^{(k)}$. Similarly, $S_A^{(k+1)}$ is a Schur complement of $S_A^{(k)}$. Next, note that for two SPD matrices A and B, $\mathbf{w}^T A \mathbf{w} \leq \mathbf{w}^T B \mathbf{w}$ for all $\mathbf{w}$ implies $\mathbf{w}^T A^{-1} \mathbf{w} \geq \mathbf{w}^T B^{-1} \mathbf{w}$ for all $\mathbf{w}$, and (using Lemma 3.1(c)) the same relations holds for their respective Schur complements. Hence, $\mathbf{v}^T S_A^{(k+1)} \mathbf{v} \leq \mathbf{v}^T S_B^{(k+1)} \mathbf{v}$. On the other hand, (3.3) implies $\mathbf{v}^T S_B^{(k+1)} \mathbf{v} \leq \mathbf{v}^T \widetilde{S}_B^{(k+1)} \mathbf{v}$, and the combination of both inequalities proves the assertion for $k + 1$.

Now, right inequality (3.34) follows from (2.5), whereas left inequality (3.35) stems from (3.33) together with left inequality (3.34) and $\|S^{1/2}\| = \|S\|^{1/2}$ for any SPD S. Eventually, the right inequality (3.35) follows from

$$\left\| S_A^{(k)\,-1} \right\| \leq \|A^{-1}\| ,$$

which itself stems from Lemma 3.1(c) . ■

4 Numerical experiments

4.1 Model problem

We consider the linear system arising from the five-point finite difference discretization of

$$\begin{aligned} -\Delta u + \varepsilon u &= f \qquad \text{in } \Omega = (0,1)^2 \\ \tfrac{\partial u}{\partial n} &= 0 \qquad \text{on } \partial\Omega \end{aligned} \tag{4.1}$$

on a uniform grid Ω_h of mesh size $h = 1/(N-1)$. Let the grid be partitioned into three disjoint subsets

$$\Omega_h^{(\mathrm{I})} = \{ \, (ih, jh) \mid 0 \leq i \leq N-1 \, , \; 0 \leq j < \lfloor N/2 \rfloor \, \}, \tag{4.2}$$

$$\Omega_h^{(\mathrm{II})} = \{ \, (ih, jh) \mid 0 \leq i \leq N-1 \, , \; \lfloor N/2 \rfloor < j \leq N-1 \, \} , \tag{4.3}$$

$$\Omega_h^{(\Gamma)} = \{ \, (ih, jh) \mid 0 \leq i \leq N-1 \, , \; j = \lfloor N/2 \rfloor \, \} , \tag{4.4}$$

such that the first two are disconnected. Assuming the lexicographical ordering of unknowns inside subsets and ordering those in $\Omega_h^{(\mathrm{I})}$ first, those in $\Omega_h^{(\mathrm{II})}$ next, and those in $\Omega_h^{(\Gamma)}$ last, the $N^2 \times N^2$ system matrix is given by

$$A_\Omega \;=\; \begin{pmatrix} A_{\mathrm{I}} & & A_{\mathrm{I},\Gamma} \\ & A_{\mathrm{II}} & A_{\mathrm{II},\Gamma} \\ A_{\mathrm{I},\Gamma}^T & A_{\mathrm{II},\Gamma}^T & A_\Gamma \end{pmatrix}. \tag{4.5}$$

For $\epsilon > 0$ the matrix A_Ω is symmetric and strictly diagonally dominant; hence, it is SPD.

Often, the unknowns corresponding to $\Omega_h^{(\mathrm{I})}$, $\Omega_h^{(\mathrm{II})}$ are further eliminated. This happens, for instance, prior to the last stage of the (exact) Cholesky factorization method based on nested dissection [11, 19]. The $N \times N$ system matrix

$$A \;=\; A_\Gamma - A_{\mathrm{I},\Gamma}^T A_{\mathrm{I}}^{-1} A_{\mathrm{I},\Gamma} - A_{\mathrm{II},\Gamma}^T A_{\mathrm{II}}^{-1} A_{\mathrm{II},\Gamma} \tag{4.6}$$

of the resulting reduced system corresponds to the Schur complement of A_Ω with respect to its bottom rightmost block. It follows from Lemma 3.1(b) that A is also SPD if $\epsilon > 0$.

Note that A is usually not sparse and its Cholesky factorization requires $\mathcal{O}(N^3)$ operations. An iterative solution may therefore be an attractive alternative for large N. In particular, the HSS and SSS variants of Algorithm 2.2 as described in Section 2.3 only require $\mathcal{O}(r_{\max}N^2)$ operations to construct the preconditioner $B_\ell = R^T R$ of A and, as we shell see below, the maximal rank $r_{\max}$ in the approximations remains bounded (or, at least, grows slowly with N). Hence, if the condition number of the resulting system is bounded as well, the overall complexity is also[1] $\mathcal{O}(r_{\max}N^2)$. Note that the preconditioner for the original (i.e., non reduced) system may be chosen as $B_\Omega = R_\Omega^T R_\Omega$, where

$$R_\Omega \;=\; \begin{pmatrix} R_{\mathrm{I}} & & R_{\mathrm{I}}^{-T} A_{\mathrm{I},\Gamma} \\ & R_{\mathrm{II}} & R_{\mathrm{II}}^{-T} A_{\mathrm{II},\Gamma} \\ & & R \end{pmatrix},$$

with R_{I}, R_{II} being upper triangular and such that $R_{\mathrm{I}}^T R_{\mathrm{I}} = A_{\mathrm{I}}$, $R_{\mathrm{II}}^T R_{\mathrm{II}} = A_{\mathrm{II}}$. In this case

$$\kappa(B_\Omega^{-1} A_\Omega) \;=\; \kappa(R^{-T} A R^{-1})$$

and, hence, our conditioning analysis for the reduced system also applies to the original one.

[1] This comes with the fact that one application of the preconditioner requires at most $\mathcal{O}(N^2)$, even for the exact Cholesky factor. For HSS and SSS variants the complexity is linear in N.

4.2 Fixed threshold experiments

We now investigate how accurately the bounds derived in Section 3 reproduce the condition number of the preconditioners described by Algorithm 2.2 . More precisely, we consider the HSS, SSS and one-level variants of the algorithm, as well as the algorithm from [27]; APPROX$(\cdot)$ operation corresponds in all cases to the truncated SVD decomposition with relative threshold tol_r, as presented in Section 2.2.

The matrix A is subdivided into the $n \times n$ block form (2.1), with block size $|\mathcal{I}_i| = 10$, $i = 1,..,n$ and, hence, with $N = 10n$. Next, to use HSS variant with binary trees as described in Section 2.3, we set $n = 2^t + 1$, where t is the tree depth. Here we consider t from 0 to 6, which corresponds to matrix sizes N ranging from 20 to 650; the size N^2 of the unreduced matrix A_Ω is then between 400 and 422500 . For every such N^2 we list in Table 1 the number of approximation steps ℓ performed during the factorization; this number is approximately two times larger for HSS variant an the method in [27] compared to SSS and one-level variants. Further, we are interested in matrices A that are ill-conditioned; to achieve this we set $\varepsilon = 10^{-4}$, with $\kappa(A)$ then ranging from 1.2 10^6 to 3.7 10^7 .

Now, we report on Figure 4 for different values of unreduced system size N^2 and of relative dropping threshold tol_r the exact condition number for the considered preconditioners as well as the corresponding upper bounds. More precisely, the bound for the one-level variant is computed by repeatedly applying (3.6), (3.5) with $p = \ell - k$, whereas (3.16) is used for the SSS variant and Algorithm 3.1 for the HSS one. In all cases, the values of γ_k , $k = 1, ..., \ell$, are computed using the definition (3.7). The figure also highlight the parameters $\overline{\gamma}\ell$ and $\widetilde{\gamma}^2\ell$ for different tol_r and N^2 . Maximal and minimal ranks for a given threshold are reported in Table 2 .

First, we note that the condition numbers of all the preconditioners remain close to each other; for the same problem size and threshold value they differ at most by a factor of 11 (this factor reduces to 5 for the largest size). Note that the method from [27] achieves this with the double of the ranks values in the other cases; this is mainly due to the fact that the matrices compressed effectively during each step are then larger than in the other cases. We also note that these results are analogous to what is usually observed for similar problems with multigrid methods [23, 18], for which a simple two-grid scheme (analog of a one-level variant here) behaves similarly to more practical multigrid methods (alike SSS/HSS/[27] variants here).

Second, the one-level estimate follows closely the corresponding condition number, even for large values of N^2, where the number ℓ of approximation steps is large. On the other hand, the estimates for the other two approaches are less accurate, especially when the corresponding condition number is away from 1. Note that

N^2	400	900	2500	8100	28900	108900	422500
one-level/SSS	1	2	4	8	16	32	64
HSS/[27]	1	3	7	15	31	63	127

Table 1: Number ℓ of approximation steps.

	one-level/SSS/HSS		method in [27]	
tol_r	10^{-1}	10^{-3}	10^{-1}	10^{-3}
rank	2–2	3–5	2–4	3–10

Table 2: Maximal and minimal ranks for all considered values of N.

the HSS bound obtained with Algorithm 3.1 is close to the SSS bound, despite the higher number ℓ of approximation steps; this comes with additional assumptions in HSS case on the indices of the modified rows. Eventually, pushing further the multigrid analogy, we note that the one-level condition estimate reproduce correctly the convergence behavior of the other approaches, and therefore may be used to estimate their convergence rate; it is similar to the two-grid analysis in this respect.

Now, regarding the accuracy parameters $\overline{\gamma}\ell = \sum_{k=1}^{\ell} \gamma_k$ and $\widetilde{\gamma}^2\ell = \sum_{k=1}^{\ell} \gamma_k^2$ we note that, as suggested by the analysis in Section 3 (see the comments for Corollaries 3.4 and 3.6), the condition numbers and the related estimates of, respectively, SSS/HSS and one-level variants remain nicely bounded if these parameters remains below or around 1. The converse seems also true for $\widetilde{\gamma}^2\ell$; namely, the values of this parameter substantially larger than 1 come with the high values of all upper bounds, but also of all the condition numbers.

4.3 Adaptive threshold strategies

Another observation illustrated in Figure 4 is that the condition numbers, their estimates and the corresponding accuracy parameters $\overline{\gamma}\ell$, $\widetilde{\gamma}^2\ell$ tend to increase with N, independently of the value tol_r of the truncation threshold. In the case of accuracy parameters, this grows has two contributions: the increasing number ℓ of the approximation steps as well as the increase in the norm of $\widetilde{S}_B^{(k)}$ which enters the definition (3.7) of γ_k.

Let us now determine the conditions under which the condition number remains bounded independently of N. Clearly, if we require

$$\gamma_k \le \frac{c}{\ell}, \tag{4.7}$$

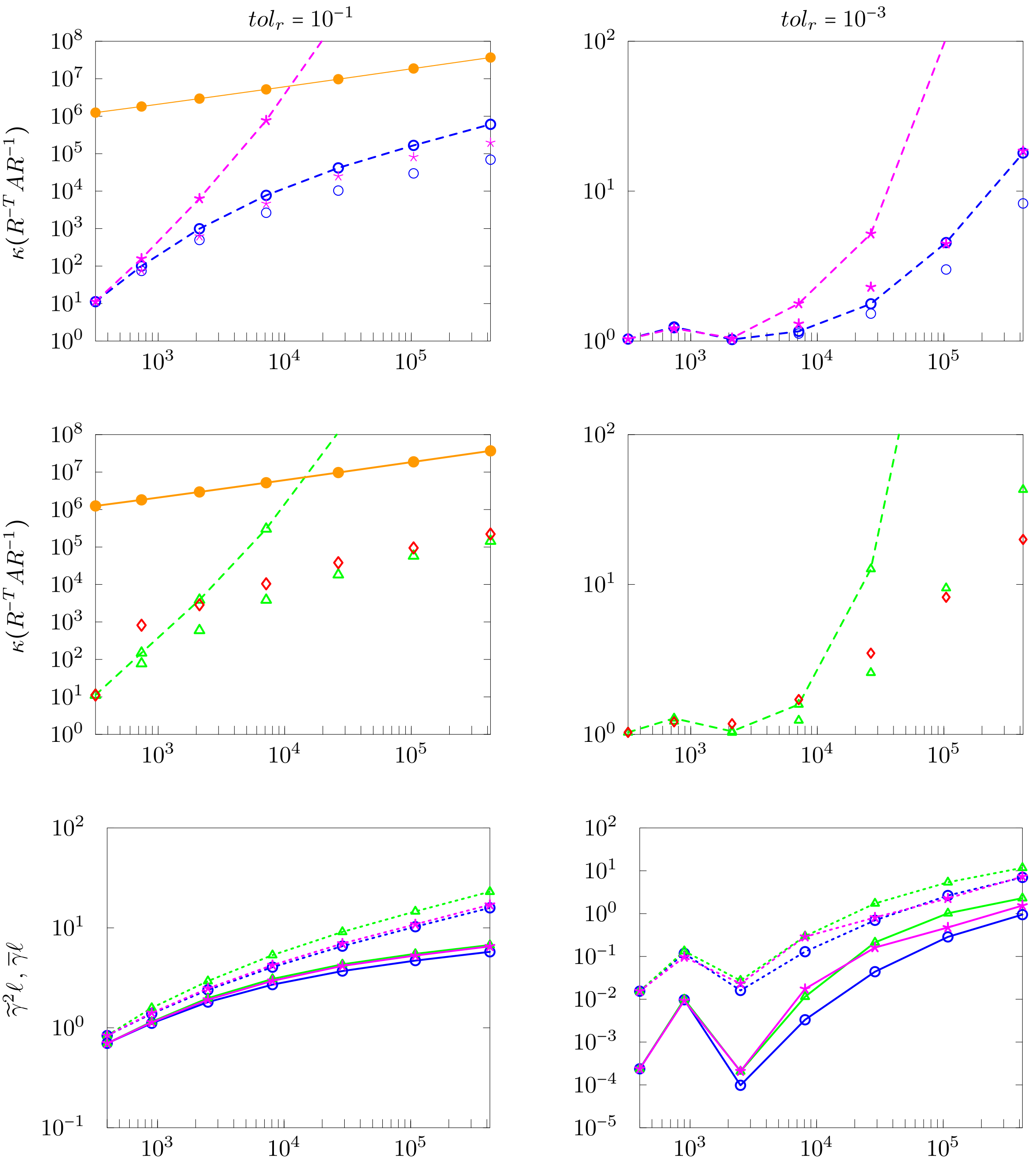

Figure 4: Condition number $\kappa(R^{-T}AR^{-1})$ for various strategies and the related upper bounds (top and middle), together with the corresponding accuracy parameters $\overline{\gamma}\ell, \widetilde{\gamma}^2\ell$ (bottom) for truncation threshold tol_r set to 10^{-1} (left) and 10^{-3} (right) and for different values of N^2. Markers ○ (blue) corresponds to one-level variant, ★ (magenta) to SSS, △ (green) to HSS, and ◇ (red) to [27]. On the top and middle plots, isolated markers correspond to $\kappa(B^{-1}A)$, dashed lines connect their upper bounds and ● (orange) connected by a solid line represent $\kappa(A)$. On the bottom plots, dotted lines depict $\overline{\gamma}\ell$ (see Corollary 3.4) and solid ones stand for $\widetilde{\gamma}^2\ell$ (see Corollary 3.6).

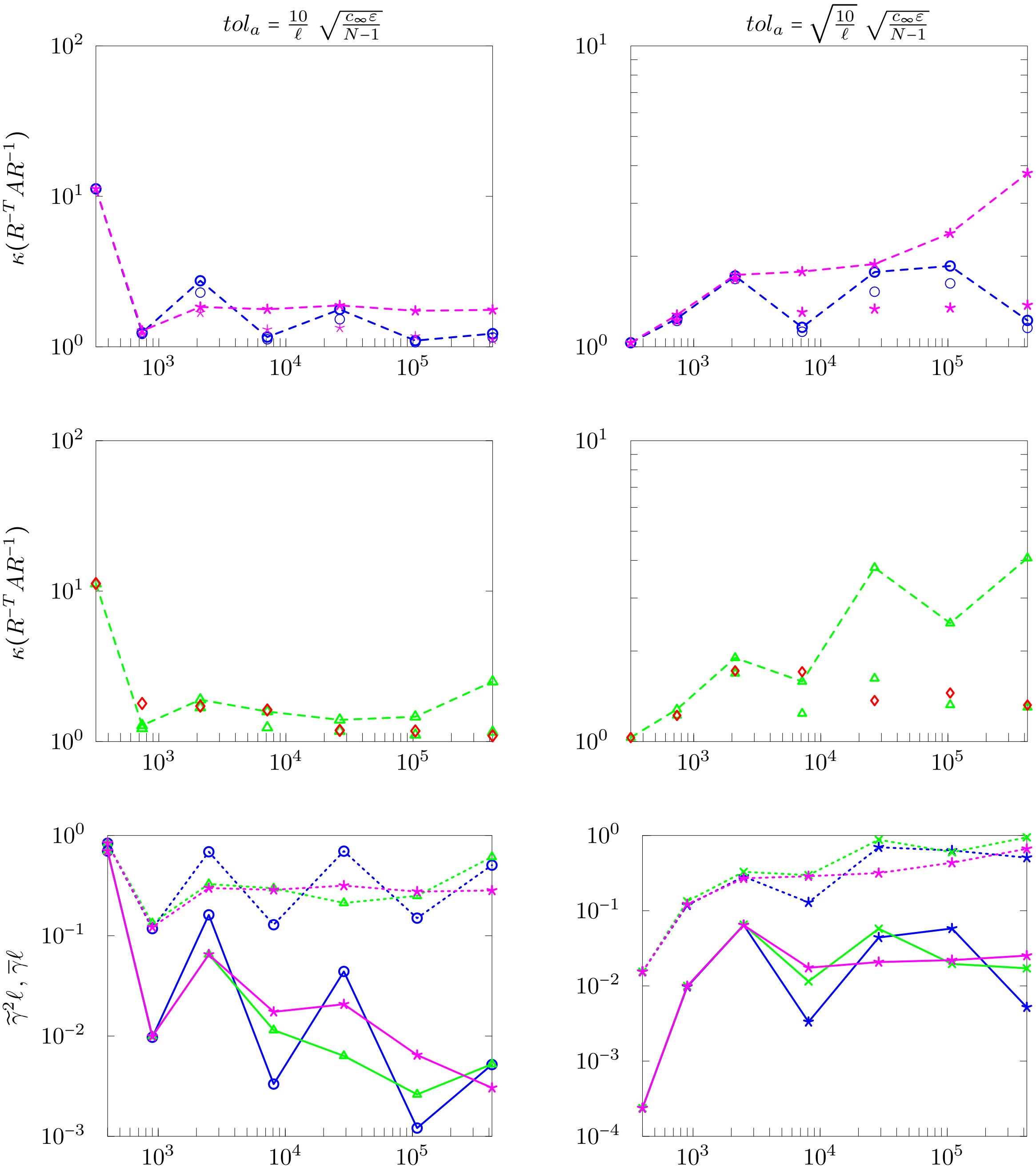

Figure 5: Condition numbers $\kappa(R^{-T}AR^{-1})$, their upper bounds (top and center) and the corresponding accuracy parameters $\overline{\gamma}\ell, \widetilde{\gamma}^2\ell$ (bottom) for tol_a given by (4.9) (left) and (4.10) (right).

for some $c < 1$, then $\overline{\gamma}\ell \le c$ and the SSS estimate (3.17) will remain bounded. As mentioned in Section 3, relaxing the requirement on c to $c = \mathcal{O}(1)$ should still guarantee a bounded condition number for all the considered methods. Now, to estimate γ_k we use right inequality (3.35) where, as shown in Appendix A,

$$\|A^{-1}\| \;\le\; c_N^{-1} \cdot \frac{N-1}{\varepsilon} \tag{4.8}$$

with $c_N \to \frac{1}{2}e^{-4\sqrt{\varepsilon}}$ for $N \to \infty$; for simplicity, we use c_∞ instead of c_N. Hence, combining (3.35), (2.12) with the above inequalities (4.7), (4.8), and using the fact that $\ell = \mathcal{O}(N)$ further yields

$$tol_a \;\le\; \frac{c}{\ell}\sqrt{\frac{c_\infty \varepsilon}{N-1}} \;=\; \mathcal{O}(N^{-3/2})\,. \tag{4.9}$$

Note the use of absolute threshold here, as opposite to a more common relative threshold in the previous subsection.

On the other hand, as may be concluded from the previous subsection, the one-level bound based on the repeated application of (3.6), (3.5) with $p = \ell - k$ provides an accurate condition number estimate for the considered methods. In this case, it is for instance enough to require $\widetilde{\gamma}^2\ell \;\le\; c$, as follows from Corollary 3.6; that is,

$$\gamma_k \;\le\; \sqrt{\frac{c}{\ell}}$$

must hold for some $c = \mathcal{O}(1)$. Combining this latter with (3.35), (2.12) and (4.8) entails a less restrictive condition

$$tol_a \;\le\; \sqrt{\frac{c}{\ell}}\sqrt{\frac{c_\infty \varepsilon}{N-1}} \;=\; \mathcal{O}(N^{-1})\,. \tag{4.10}$$

Now, the results for both strategies are given for $c = 10$ on Figure 5. Note that both strategies are effective in keeping the condition number bounded above. Since the strategy based on controlling $\widetilde{\gamma}^2\ell$ is less restrictive, it should be preferred.

5 Concluding remarks

We have presented a conditioning analysis of incomplete Cholesky factorizations based on orthogonal dropping. The analysis covers several existing preconditioners and provides an upper bound which only depends on the accuracy γ_k of individual approximations. Whereas no assumption on the indices of rows modified during each approximation step is required for the analysis to hold, such assumptions may further improve the resulting estimate.

Now, the best improvement is obtained for the preconditioners based on the one-level index choice. The corresponding bound is further shown sharp for any possible set γ_k, $k = 1, ..., \ell$, of accuracy measures. Moreover, numerical experiments reveal that one-level bound allows an accurate estimation of the condition number for various index choices, including one-level, SSS and HSS.

Regarding the accuracy measure γ_k, one may estimate its value (as shown in Theorem 3.8) by assessing the norm $\|R_{12}^{(k)} - \widetilde{R}_{12}^{(k)}\|$ of the dropped component and, for instance, the norm $\|A^{-1}\|$ of the inverse of the system matrix. The later parameter may be obtained with few iterations of the conjugate gradient method, whereas the former is directly controlled via the threshold value tol_a of a truncated orthogonal decomposition. Hence, the analysis offers a practical way to control the condition number of the resulting preconditioners. The potentialities of such approach are highlighted in our numerical experiments with adaptive threshold strategies. We do not pursue this discussion here, however, since it is subject to further research.

Acknowledgment

I thank Xiaoye S. Li and Yvan Notay for their comments on the preliminary version of this manuscript.

Appendix A

Here we show that

$$A \;\geq\; c_N \cdot \frac{\varepsilon}{N-1} I \, ,$$

where $\lim_{N\to\infty} c_N \;=\; \frac{1}{2}e^{-4\sqrt{\varepsilon}}$ and where the Schur complement A is defined by (4.6), with A_Ω of the form (4.5) corresponding to the five-point discretization of the boundary value problem (4.1) with $\varepsilon \;\leq\; 1$. First, for the considered discretization one has

$$A_m \;=\; \begin{pmatrix} \frac{1}{2}T_0 & -T_1 & & \\ -T_1 & T_0 & \ddots & \\ & \ddots & \ddots & -T_1 \\ & & -T_1 & T_0 \end{pmatrix} , \quad A_{m,\Gamma} \;=\; \begin{pmatrix} \\ \\ \\ -T_1 \end{pmatrix} , \quad A_\Gamma = T_0 \, , \quad m \;=\; \mathrm{I}, \mathrm{II} \, ,$$

where $T_1 \;=\; \frac{1}{2}\mathrm{diag}(1, 2, ..., 2, 1)$ and $T_0 \;=\; (4 + \varepsilon_N)T_1 + \mathrm{tridiag}(-1 \quad 0 \quad -1)$, $\varepsilon_N \;=\; \varepsilon \;(N-1)^{-2}$. Note that T_0, T_1 are SPD and satisfy

$$\frac{1}{2}I \;\leq\; T_1 \;\leq\; I \, , \tag{A.1}$$

$$T_0 \;\geq\; (2 + \varepsilon_N)T_1 \, , \tag{A.2}$$

both inequalities (as well as the matrix inequalities below) holding in SPD sense; in what follows we use only these inequalities, not the matrices T_0, T_1 themselves.

We begin with the observation that

$$A_{m,\Gamma}^T A_m^{-1} A_{\Gamma,m} \;=\; T_1 S_{s_m}^{-1} T_1\,, \quad m = \mathrm{I}, \mathrm{II}\,, \tag{A.3}$$

where $s_{\mathrm{I}} = \lfloor N/2 \rfloor$ and $s_{\mathrm{II}} = N - \lfloor N/2 \rfloor - 1$ are the block dimension of A_{I}, A_{II} respectively and where S_{s_m} is the Schur complement of the right bottom block of $A_{m,\Gamma}$ that satisfy

$$S_i \;=\; T_0 - T_1 S_{i-1}^{-1} T_1\,, \quad S_0 \;=\; \frac{1}{2} T_0\,. \tag{A.4}$$

The latter recursion may be obtained, for instance, by repeated application of Lemma 3.1(c). Together with (4.6) it further implies

$$A \;=\; T_0 - T_1 (S_{s_{\mathrm{I}}}^{-1} + S_{s_{\mathrm{II}}}^{-1}) T_1\,. \tag{A.5}$$

Now, we show the desired result by deriving a lower bound on S_{s_m}, $m = \mathrm{I}, \mathrm{II}$ and using it in (A.5). First, note that S_i as defined by (A.4) is an increasing function of T_0; hence, it follows from (A.2) that, setting

$$\widetilde{S}_i \;=\; (2 + \varepsilon_N) T_1 - T_1 \widetilde{S}_{i-1}^{-1} T_1\,, \quad \widetilde{S}_0 \;=\; \frac{1}{2}(2 + \varepsilon_N) T_1\,, \tag{A.6}$$

there holds $S_{s_m} \;\geq\; \widetilde{S}_{s_m}$. Next, one has

$$\widetilde{S}_1 - \widetilde{S}_0 \;=\; \left(\frac{2 + \varepsilon_N}{2} - \frac{2}{2 + \varepsilon_N}\right) T_1 \;=\; \bar{c}_N\, \varepsilon_N T_1\,. \tag{A.7}$$

with $\bar{c}_N = (4 + \varepsilon_N)/(4 + 2\varepsilon_N) \to 1$ as $N \to \infty$. Hence, $\widetilde{S}_1 \geq \widetilde{S}_0$ and further, by recursive application of (A.6), there holds

$$\widetilde{S}_\infty \;\geq\; \cdots \;\geq\; \widetilde{S}_i \;\geq\; \cdots \;\geq\; \widetilde{S}_0 \;=\; \frac{1}{2}(2 + \varepsilon_N)\,, \tag{A.8}$$

where $\widetilde{S}_\infty$ exists (the sequence $\widetilde{S}_i$ is increasing and bounded above by $(2 + \varepsilon_N) T_1$, as follows from (A.6)) and satisfy

$$S_\infty \;=\; (2 + \varepsilon_N) T_1 - T_1 S_\infty^{-1} T_1\,.$$

Hence,

$$\widetilde{S}_\infty \;=\; \frac{(2 + \varepsilon_N) + \sqrt{(2 + \varepsilon_N)^2 - 4}}{2} T_1 \;\leq\; \left(1 + \frac{1 + \sqrt{5}}{2} \sqrt{\varepsilon_N}\right) T_1 \;\leq\; \left(1 + 2\sqrt{\varepsilon_N}\right) T_1\,,$$

where the first inequality follows from $\varepsilon_N^2 \le \varepsilon_N \le \sqrt{\varepsilon_N} \le 1$. Now, using this latter together with (A.8), (A.6) and the fact that $\widetilde{S}_i \le \widetilde{S}_\infty$ has the same eigenbasis as T_1 further entails

$$\widetilde{S}_i - \widetilde{S}_{i-1} \;=\; T_1(\widetilde{S}_{i-2}^{-1} - \widetilde{S}_{i-1}^{-1})T_1 \;\ge\; T_1\widetilde{S}_{i-1}^{-1}(\widetilde{S}_{i-1} - \widetilde{S}_{i-2})\widetilde{S}_{i-1}^{-1}T_1 \;\ge\; \frac{1}{(1+2\sqrt{\varepsilon_N})^2}(\widetilde{S}_{i-1} - \widetilde{S}_{i-2})\,.$$

Repeated application of this latter in combination with (A.7) gives

$$\widetilde{S}_i - \widetilde{S}_{i-1} \;\ge\; \bar{c}_N \frac{\varepsilon_N}{(1+2\sqrt{\varepsilon_N})^{2i-2}} T_1\,.$$

Further, adding up the above contributions and using

$$e^{4i\sqrt{\varepsilon_N}} \;\ge\; (1+2\sqrt{\varepsilon_N})^{2i} \;\ge\; 1 + 4i\sqrt{\varepsilon_N}$$

yields

$$\begin{aligned}
\widetilde{S}_i \;&\ge\; \widetilde{S}_0 + (\widetilde{S}_1 - \widetilde{S}_0) + \cdots + (\widetilde{S}_i - \widetilde{S}_{i-1}) \\
&\ge\; T_1 + \bar{c}_N\varepsilon_N \left(1 + \frac{1}{(1+2\sqrt{\varepsilon_N})^2} + \cdots + \frac{1}{(1+2\sqrt{\varepsilon_N})^{2i-2}}\right) T_1 \\
&= T_1 + \frac{\bar{c}_N\varepsilon_N}{(1+2\sqrt{\varepsilon_N})^{2i-2}} \frac{(1+2\sqrt{\varepsilon_N})^{2i} - 1}{(1+2\sqrt{\varepsilon_N})^2 - 1}\, T_1 \\
&=\; T_1 + \frac{\bar{c}_N\sqrt{\varepsilon_N}}{(1+2\sqrt{\varepsilon_N})^{2i-2}} \frac{(1+2\sqrt{\varepsilon_N})^{2i} - 1}{4(1+\sqrt{\varepsilon_N})}\, T_1 \\
&\ge\; \left(1 + \frac{\bar{c}_N i\varepsilon_N}{e^{4\sqrt{\varepsilon_N}(i-1)}(1+\sqrt{\varepsilon_N})}\right) T_1\,.
\end{aligned}$$

Now, noting that $\varepsilon_N = \varepsilon\,(N-1)^{-2}$ and $i_{\mathrm{I}}, i_{\mathrm{II}} \approx N/2$, and that $\bar{c}_N \to 1$ as $N \to \infty$, one has

$$S_{i_m} \;\ge\; \widetilde{S}_{i_m} \;\ge\; \left(1 + \widetilde{c}_N \frac{\varepsilon}{N-1}\right) T_1\,, \quad m = \mathrm{I}, \mathrm{II}\,,$$

where $\widetilde{c}_N \to \frac{1}{2}e^{-4\sqrt{\varepsilon}}$ for $N \to \infty$. Together with (A.5), (A.2) this further entails

$$A \;=\; T_0 - T_1(S_{i_{\mathrm{I}}}^{-1} + S_{i_{\mathrm{II}}}^{-1})T_1 \;\ge\; (2 - \frac{2}{1+\frac{\widetilde{c}_N\varepsilon}{N-1}})T_1 \;=\; \frac{\widetilde{c}_N}{1+\frac{\widetilde{c}_N\varepsilon}{N-1}} \frac{\varepsilon}{N-1} 2T_1 \;=\; c_N \frac{\varepsilon}{N-1} 2T_1\;,$$

where $c_N \to \frac{1}{2}e^{-4\sqrt{\varepsilon}}$ for $N \to \infty$, and the result then follows from (A.1).

References

[1] O. Axelsson. *Iterative Solution Methods.* Cambridge University Press, Cambridge, 1994.

[2] O. Axelsson and I. Gustafsson. Preconditioning and two-level multigrid methods of arbitrary degree of approximation. *Math. Comp.*, 40:214–242, 1983.

[3] M. Bebendorf. Hierarchical LU decomposition-based preconditioners for BEM. *Computing*, 74:225–247, 2005.

[4] M. Bebendorf. Why finite element discretizations can be factored by triangular hierarchical matrices. *SIAM J. Numer. Anal.*, 45:1472–1494, 2007.

[5] M. Bebendorf and W. Hackbusch. Existence of $\mathcal{H}$-matrix approximants to the inverse FE-matrix of elliptic operators with L^∞-coefficients. *Numer. Math.*, 95:1–28, 2003.

[6] S. Börm. Approximation of solution operators of elliptic partial differential equations by $\mathcal{H}$- and $\mathcal{H}^2$-matrices. *Numer. Math.*, 115:165–193, 2010.

[7] P. Businger and G. H. Golub. Linear least squares solutions by householder transformations. *Numer. Math.*, 7:269–276, 1965.

[8] T. F. Chan. Rank revealing QR factorizations. *Linear Algebra Appl.*, 88/89:67–82, 1987.

[9] S. Chandrasekaran, P. Dewilde, M. Gu, and N. Somasunderam. On the numerical rank of the off-diagonal blocks of Schur complements of discretized elliptic PDEs. *SIAM J. Matrix Anal. Appl.*, 31:2261–2290, 2010.

[10] S. Chandrasekaran and I. C. F. Ipsen. On rank-revealing factorizations. *SIAM J. Matrix Anal. Appl.*, 15:592–622, 1994.

[11] A. George. Nested dissection of a regular finite-element mesh. *SIAM J. Numer. Anal.*, 10:345–363, 1973.

[12] G. H. Golub and C. F. van Loan. *Matrix Computations.* The John Hopkins University Press, Baltimore, Maryland, 1996. Third ed.

[13] L. Grasedyck, R. Kriemann, and S. Le Borne. Domain decomposition based $\mathcal{H}$-LU preconditioning. *Numer. Math.*, 112:565–600, 2009.

[14] A. Greenbaum. *Iterative Methods for Solving Linear Systems*, volume 17 of *Frontiers in Applied Mathematics.* SIAM, Philadelphia, PA, 1997.

[15] M. Gu and S. C. Eisenstat. Efficient algorithms for computing a strong-rank revealing QR factorization. *SIAM J. Sci. Comput.*, 17:848–869, 1996.

[16] M. Gu, X. S. Li, and P. Vesselevski. Direction-preserving and Schur-monotonic semiseparable approximations of symmetric positive definite matrices. *SIAM J. Matrix Anal. Appl.*, 31:2650–2664, 2010.

[17] I. Gustafsson. A class of first order factorization methods. *BIT*, 18:142–156, 1978.

[18] W. Hackbusch. *Multi-grid Methods and Applications.* Springer, Berlin, 1985.

[19] R. J. Lipton, D. J. Rose, and R. E. Tarjan. Generalized nested dissection. *SIAM J. Numer. Anal.*, 16:346–358, 1979.

[20] J. A. Meijerink and H. A. van der Vorst. An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix. *Math. Comp.*, 31:148–162, 1977.

[21] Y. Saad. ILUT: a dual threshold incomplete ILU factorization. *Numer. Lin. Alg. Appl.*, 1:387–402, 1994.

[22] Y. Saad. *Iterative Methods for Sparse Linear Systems.* SIAM, Philadelphia, PA, 2003. Second ed.

[23] U. Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid.* Academic Press, London, 2001.

[24] H. A. van der Vorst. *Iterative Krylov Methods for Large Linear systems.* Cambridge University Press, Cambridge, 2003.

[25] J. Xia, S. Chandraserkaran, M. Gu, and X. S. Li. Fast algorithms for hierarchically semiseparable matrices. *Numer. Lin. Alg. Appl.*, 17:953–976, 2009.

[26] J. Xia, S. Chandraserkaran, M. Gu, and X. S. Li. Superfast mulrtifrontal method for large structured linear systems of equations. *SIAM J. Matrix Anal. Appl.*, 31:1382–1411, 2009.

[27] J. Xia and M. Gu. Robust approximate Cholesky factorization of rank-structured symmetric positive definite matrices. *SIAM J. Matrix Anal. Appl.*, 31:2899–2920, 2010.