Operator Theory:
Advances and Applications, Vol. 197, 99–132

Model Reduction in Symbolically Semi-separable Systems with Application to Pre-conditioners for 3D Sparse Systems of Equations

Patrick Dewilde, Haiyan Jiao and Shiv Chandrasekaran

In dear memory of Prof. M. Livsic

Abstract. Preconditioned iterative solvers are considered to be one of the most promising methods for solving large and sparse linear systems. It has been shown in the literature that their impact can be fairly easily extended to semi-separable systems or even larger classes build on semi-separable ideas. In this paper, we propose and evaluate a new type of preconditioners for the class of matrices that have a two level deep 'symbolically hierarchical semi-separable form' meaning that the matrices have a semi-separable like block structure with blocks that are (sequentially) semi-separable themselves. The new preconditioners are based on approximations of Schur complements in a sequential or hierarchical decomposition of the original block matrix. The type of matrices considered commonly occur in 3D modeling problems.

Mathematics Subject Classification (2000). 65F10, 65F15; 65F50; 93A13.

Keywords. Preconditioners, semi-separable systems, model reduction, Poisson equation.

1. Introduction

The importance of preconditioners to solve large systems of sparse equations has been amply demonstrated in the literature, an excellent survey is to be found in [3]. However, in a number of crucial cases, finding good preconditioners has proved to be very difficult, if not impossible, lacking a systematic method to construct them either from basic principles or from the physical circumstances leading to the system to be solved. In this paper we propose a method based on algebraic principles, but which can also accommodate physical considerations to some extent. The proposed method is adequate to handle systems that extend in 2 dimensions

(2D-systems) but we want to show that the ideas will extent to 3D systems as well. Although the method applies to a fairly general class of systems, we are only able to validate it on systems that can be solved explicitly. In this paper we consider one type of such systems: positive definite, Hermitian of the block-Toeplitz-block-Toeplitz (BTBT) kind and we consider two cases: purely Toeplitz and circulant. Validation to larger classes has necessarily to be experimental, but the physical connotations of the method makes it a very good candidate for future use in 3D systems in general.

The generic system solver solves a set of equations

$$\Phi u = b \tag{1}$$

in which Φ is a square matrix, b a conformal vector of data and u is the solution vector to be found. We assume that Φ is non-singular and even that it can be LU-factored (the method can be extended to the Moore-Penrose case, but that is beyond our present scope). We put further assumptions on the structure of Φ that make the class considered adequate for fairly general modeling problems that lead to 3D sparse matrices.

A good preconditioner P is a matrix of the same dimensions as Φ such that (1) $I - \Phi P$ is small and (2) multiplication with P is computationally cheap. Iterative solvers are adequate when (1) also the multiplication with Φ is cheap and (2) a good P is known or can easily be determined. The iterative solver will then iterate on the error residue and converge quickly when the eigenvalues of $I - \Phi P$ are close to zero – we refer to the literature for more details [3].

Solvers based on preconditioners are obviously attractive when Φ is a sparse matrix, for then the condition of 'cheap multiplication with Φ' is automatically fulfilled. However, this is certainly not the only class that leads to cheap multiplication. Another is the class of 'sequentially semi-separable matrices' [5, 7], or the class of 'hierarchically semi-separable matrices' [1]. These classes are distinct, sparse matrices are not semi-separable in general (only banded matrices are). Hierarchically semi-separable matrices can be transformed into specific classes of sparse matrices, making it an attractive class because the extra structure allows for efficient solving, either in a direct or a preconditioned way. The problem with general classes of sparse matrices is the difficulty of finding a good preconditioner. Our approach is to extend the class of structured matrices of the semi-separable type so that it covers a wider collection of sparse matrices and transformations thereof. The extension that we consider in this paper (and that is described in the next paragraph) is able to cover most, if not all, 2D type modeling problems, whether of the sparse type or the so-called 'multipole' type.

The matrix structure that we consider in this paper can be termed 'symbolically semi-separable'. We shall treat the semi-separable structure extensively in a further section. A semi-separable matrix is characterized by a so-called 'realization', i.e., an ordered sets of seven (small) diagonal block-matrices denoted, e.g., as $\{A, B, C, D, A', B', C'\}$. We say that the structure is 'symbolical' if the characterization has the same form, but the characterizing set of matrices has further struc-

ture, namely all the submatrices are themselves either sequentially semi-separable or symbolically semi-separable again. E.g., if $A = \mathrm{diag}[\cdots A_k \cdots]$ where each A_k is sequentially semi-separable (and hence characterized again by a realization at a lower hierarchical level) then the symbolical hierarchy will have two layers.

Hence, our goal will be the construction of preconditioners, assuming the underlying matrix structure to be given in terms of blocks that themselves have a sequential or symbolical semi-separable structure. We shall formulate the theory and the results at a 'medium complexity level' – to keep things as simple as possible without endangering the generality needed to handle significant 3D modeling cases. In particular, we shall assume a block tri-diagonal form for the top level hierarchy. This structure is less general than full blown symbolic semiseparability, but it does cover the main application, namely systems originating from 3D finite element modeling. A special case is obtained when second-order 3D partial differential equation is considered on a regular (finite 3D) grid. We shall develop this case for Laplace's (or Poisson's) equation in the next section and carry it as a test case throughout the paper, comparing the performance of the various preconditioners proposed. In particular, we use a 27-point stencil to discretized the PDF, basic cells of dimension 8×8 resulting in an overall matrix of dimension $8^3 \times 8^3$. Measures for performance of the preconditioner P are norm differences between I and ΦP and the largest eigenvalue of the matrix $I - \Phi P$ because it determines the rate of convergence (we wish it typically smaller than 0.1).

2. Prototype example

As prototype example and to fix ideas, we consider Poisson's equation in a homogeneous medium, discretized on a uniform 3D grid. A formulation of Poisson's equation requires the solution of

$$-\left(\frac{\partial^2}{\partial x}u(x,y,z) + \frac{\partial^2}{\partial y}u(x,y,z) + \frac{\partial^2}{\partial z}u(x,y,z)\right) = f(x,y,z)$$

for $(x, y, z) \in \Omega$ where $\Omega = [0,1] \times [0,1] \times [0,1]$ with boundary conditions that after discretization with a 27 point stencil results in either a hierarchical $n^3 \times n^3$ block-tridiagonal block-Toeplitz or block circulant system of equations. Let us define a parameter $\epsilon = 0$ for the block-tridiagonal case and $\epsilon = 1$ for the circulant case, then the discretized equations to be solved take the form

$$\Phi u = b \tag{2}$$

$$\begin{pmatrix} M & -L^H & & & -\epsilon L \\ -L & M & -L^H & & \\ & -L & M & \ddots & \\ & & \ddots & \ddots & -L^H \\ -\epsilon L^H & & & -L & M \end{pmatrix} \begin{pmatrix} u_0 \\ u_0 \\ \vdots \\ u_{m-1} \end{pmatrix} = \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_{m-1} \end{pmatrix}$$

where we have assumed that u_i are the discretized unknowns along the ith column of the $n \times n \times n$ grid. Φ is a symmetric positive definite matrix with n block columns, and has the same sub-blocks on each of the tri-diagonal given by

$$M = \begin{pmatrix} O & -P^H & & & -\epsilon P \\ -P & O & -P^H & & \\ & -P & O & -P^H & \\ & & \ddots & \ddots & \ddots \\ -\epsilon P^H & & & -P & O \end{pmatrix} \tag{3}$$

$$L = \begin{pmatrix} R & Q^H & & & \epsilon Q \\ Q & R & Q^H & & \\ & Q & R & Q^H & \\ & & \ddots & \ddots & \ddots \\ \epsilon Q^H & & & Q & R \end{pmatrix} \tag{4}$$

$$O = \frac{1}{30} \begin{pmatrix} 128 & -14 & & & -14\epsilon \\ -14 & 128 & -14 & & \\ & -14 & 128 & -14 & \\ & & \ddots & \ddots & \ddots \\ -14\epsilon & & & -14 & 128 \end{pmatrix} \tag{5}$$

$$P = \frac{1}{30} \begin{pmatrix} 14 & 3 & & & \epsilon 3 \\ 3 & 14 & 3 & & \\ & 3 & 14 & 3 & \\ & & \ddots & \ddots & \ddots \\ \epsilon 3 & & & 3 & 14 \end{pmatrix} \tag{6}$$

$$Q = \frac{1}{30} \begin{pmatrix} 3 & 1 & & & \epsilon 1 \\ 1 & 3 & 1 & & \\ & 1 & 3 & 1 & \\ & & \ddots & \ddots & \ddots \\ \epsilon 1 & & & 1 & 3 \end{pmatrix} \tag{7}$$

$$R = \frac{1}{30} \begin{pmatrix} 14 & 3 & & & \epsilon 3 \\ 3 & 14 & 3 & & \\ & 3 & 14 & 3 & \\ & & \ddots & \ddots & \ddots \\ \epsilon 3 & & & 3 & 14 \end{pmatrix} \tag{8}$$

The example exhibits a strong hierarchical structure. At the top level we have a tri-diagonal or circulant block structure, whereby each of the component blocks again has a tri-diagonal or circulant block structure of scalar entries. The overall resulting matrix is therefore very sparse with a sparsity pattern characterized by small bunches of non-diagonals clustered in bands. Such a situation is typical for

3D systems in which there is only local interaction between the quantities (as is the case with a differential equation). The regularity produces a Toeplitz or at least a block-Toeplitz structure, but in the more general case the sparsity pattern keeps the same general structure in which many diagonals are zero, with big gaps between significant diagonals. It is those big gaps that make the elimination procedures tricky because of the systematic occurrence of fill ins in the gaps. In the next section we propose a strategy that consists in forcing only partial or approximated elimination steps so that an explosion of fill ins is avoided and replaced by approximations based on a small amount of data.

3. The basic procedure: decoupling

The preconditioners we propose in this paper are based on partitioning the set of equations and decoupling them by estimating (approximating rather than calculating) the perturbation one set exerts on the other. This approach is somewhat similar to what has been termed incomplete LU factorization in the literature [9]. The difference with this traditional ad hoc approach is in how the perturbation is gauged. An efficient realization of the perturbed matrix (actually a Schur complement) is the key in the reduced modeling. In this section we review the basis for the decoupling strategy and introduce some notation that will allow for hierarchical recursion of the procedure.

Assume that we split the set of unknowns $u \in \mathbf{V}$ into two nonintersecting subsets $u_1 \in \mathbf{V}^{(1)}$ of size n_1 and $u_2 \in \mathbf{V}^{(2)}$ of size n_2, $\mathbf{V}^{(1)} \cap \mathbf{V}^{(2)} = \emptyset$ and $n = n_1 + n_2$, as

$$u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}.$$

This splitting induces in a natural way a 2-by-2 block splitting of the matrix Φ,

$$\Phi = \begin{pmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{21} & \Phi_{22} \end{pmatrix}.$$

Then the matrix can be decomposed into a two level structure by a block LU factorization,

$$\begin{pmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{21} & \Phi_{22} \end{pmatrix} = \begin{pmatrix} I & 0 \\ \Phi_{21}\Phi_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \Phi_{11} & \Phi_{12} \\ 0 & S \end{pmatrix}$$

where I and 0 are generic identity and zero matrices of appropriate dimensions and

$$S = \Phi_{22} - \Phi_{21}\Phi_{11}^{-1}\Phi_{12}$$

is the Schur complement of Φ_{11} in Φ (as stated already, we assume existence of all relevant Schur complements).

Suppose the right-hand side vector b partitioned as above then the linear system decouples in two systems of reduced dimensions

$$\begin{aligned} \Phi_{11}\ u_1' &= b_1' \\ S\ u_2' &= b_2' \end{aligned} \tag{9}$$

with $b_1' = b_1$, $b_2' = b_2 - \Phi_{22}\Phi_{11}^{-1}b_1$, $u_1 = u_1' - \Phi_{11}^{-1}\Phi_{12}u_2'$ and $u_2 = u_2'$.

Our strategy for preconditioning consists in setting up a recursive schema of partitioning the variables and then decoupling the respective linear systems, and we do this not only at the top level of the hierarchy (as is discussed here), but recursively at lower levels as well. At each step in the procedure we approximate (or if one wishes, model reduce) the Schur complement systematically. The motivation for this is that the determination of the Schur complement is the step in the procedure where the fill ins are produced and the model complexity of the system hence increases. In many cases (and in particular the model case we are considering) approximating at this point is both physically and numerically justifiable, provided the partitioning is done in a justifiable way.

The recursive procedure can be set up in either a linear or a hierarchical manner. The linear recursion is of course the same as in the common LU factorization. In the block tri-diagonal case it reduces to a recursive determination of Schur complements, e.g., in the kth step written as

$$\begin{cases} S_0 = M_0, \\ S_{k+1} = M_{k+1} - L_k S_k^{-1} L_k^H. \end{cases} \tag{10}$$

In our model case, the recursion starts out with a block tridiagonal matrix. In the 2D case each of these blocks is again a tri-diagonal matrix. After the first step, the Schur complement then already has nine diagonals and at every step the number more than doubles, filling up the matrix quickly. It is not difficult to show that also the more general 'degree of semi-separability' [4] increases at the same rate, but at the same time it can be shown that there is a system with a low degree of semi-separability close by in operator norm. It is this model reduction that allows the determination of a low complexity approximant (in the semi-separable sense) in the 2D case. In the 3D case, however, one more level of hierarchy has to be dealt with – we discuss how to do this further on.

As discussed in the previous paragraph, a partitioning of the network (data and unknowns) leads to decoupling. This procedure can of course be repeated on each of the two sets, and then again, leading to a hierarchical decomposition tree representing the partitioning (still at this top level of the original hierarchy). Attached to each node of the tree there is the decoupled system of equations (and, of course, the corresponding primed and unprimed data sets which can be converted to each other according to the elimination formulas of the previous section). We use a level ordering notation as in the papers on 'HSS = Hierarchical Semi Separable' decompositions: the ordered index pair (k, ℓ) indicates node ℓ at level k ($\ell \in (1 \cdots 2^k)$). Node (k, ℓ), if it is not a leaf node, gets decomposed in two

nodes $(k+1, 2\ell-1)$ and $(k+1, 2\ell)$. To such a level decomposition there is a four block decomposition of the system attached to the node being decomposed. The decoupled system attached to the uneven child node is the 11 block of the parent system, while the system attached to the even child node is the Schur complement of that 11 block within the system defined by the parent node. In the sequel we shall mark the (eventually approximate) Schur complements with the index pairs indicating the level at which they define the decoupled system.

Because of the block triangular structure of the original system (and eventual semi separable generalizations thereof not considered here) there is a further hierarchical relation between Schur complements at various levels of the hierarchy.

Let Φ_α be a block triangular matrix at any given level $\alpha < \log m$ higher than the bottom level, dropping the index α for a simple notation and applying the two-by-two LU factorization on Φ we obtain Φ_{11} and Φ_{22} as block triangular matrices and Φ_{12} and Φ_{21} as low rank matrices with only one block at the left lower corner and the right upper corner respectively.

$$\Phi = \begin{pmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{21} & \Phi_{22} \end{pmatrix}.$$

Factorizing the matrix one more level we get:

$$\Phi = \begin{pmatrix} \Phi_{11_{11}} & \Phi_{11_{12}} & 0 & 0 \\ \Phi_{11_{21}} & \Phi_{11_{22}} & \Phi_{12_{21}} & 0 \\ 0 & \Phi_{21_{12}} & \Phi_{22_{11}} & \Phi_{22_{12}} \\ 0 & 0 & \Phi_{22_{21}} & \Phi_{22_{22}} \end{pmatrix}$$

with Φ_{11} and Φ_{22} the matrix decomposition for the next level. Then the Schur complement S of matrix Φ_{11} in Φ is

$$\begin{aligned} S &= \Phi_{22} - \Phi_{21}\Phi_{11}^{-1}\Phi_{12} \\ &= \begin{pmatrix} \Phi_{22_{11}} - \Phi_{21_{12}}(\Phi_{11}^{-1})_{22}\Phi_{12_{21}} & \Phi_{22_{12}} \\ \Phi_{22_{21}} & \Phi_{22_{22}} \end{pmatrix}. \end{aligned}$$

Let us put temporarily

$$E = \Phi_{11}$$

then with the two-by-two factorization block structure we find

$$E^{-1} = \begin{pmatrix} E_{11}^{-1} + E_{11}^{-1}E_{12}S_E^{-1}E_{21}E_{11}^{-1} & -E_{11}^{-1}E_{21}S_E^{-1} \\ -S_E^{-1}E_{21}E_{11}^{-1} & S_E^{-1} \end{pmatrix},$$

which shows that

$$(E^{-1})_{22} = S_E^{-1}.$$

We substitute

$$(\Phi_{11}^{-1})_{22} = S_{\Phi_{11}}^{-1}$$

back, where $S_{\Phi_{11}}$ is the Schur complement of block $\Phi_{11_{11}}$ in Φ_{11}. Therefore the Schur complement S of Φ_{11} in Φ becomes:

$$S = \begin{pmatrix} \Phi_{22_{11}} - \Phi_{21_{12}}S_{\Phi_{11}}^{-1}\Phi_{12_{21}} & \Phi_{22_{12}} \\ \Phi_{22_{21}} & \Phi_{22_{22}} \end{pmatrix}$$

in which the (11)-entry is actually the Schur complement of S_E in the submatrix

$$\begin{pmatrix} S_E & \Phi_{12_{21}} \\ \Phi_{21_{12}} & \Phi_{22_{11}} \end{pmatrix}.$$

Hence, the higher level Schur complement is constituted of lower level Schur complements and other lower level matrices.

In this fashion, the causality relations get to be very simple when the recursion is spun out to the bottom decomposition level, no calculations are needed to move to higher levels in the tree, only assembly of submatrices. In an exact calculation, a chain of Schur complements only involving local matrices is obtained at the bottom level of the hierarchy – see Fig. 3 – typically the level of the size of the block entries in the original tri-diagonal matrix, but any higher level may serve as bottom level just as well. The recursion starts out with a tridiagonal matrix and then doubles in theoretical semi-separable complexity at each step. Keeping this complexity increase under control is the key to the systematic construction of preconditioners based on Schur complementation. That is the topic of Section 7.

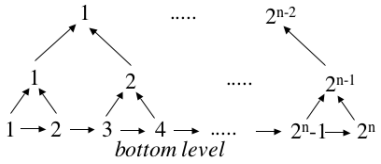

FIGURE 1. Causality relations for the Schur elimination schema. Only the bottom line requires computations, all the upward arrows only involve assembly of matrices.

4. Algorithms for sequentially semi-separable matrices

In this section we treat the case in which the Schur complement is approximated by a low-order semi-separable representation. This would be the method to be followed in the 2D case, or at the bottom level of the hierarchy in the 3D case. In [4, 7] it is shown that subsequent Schur complements occurring in the solution of the regular 2D Poisson problem are close to a low degree semi-separable matrix. In particular, in [7] the convergence in terms of the ϵ-rank of Hankel blocks is shown to be bounded with some low bound. These bounds are confirmed by experiments, which actually exhibit very close approximation even at low semi separable degree. The precise results for the regular Poisson equation are of course due to the fact that in this case the system can be solved in closed form. We start out with a brief summary of the basic properties of semi-separable representations we use. For a comprehensive treatment of the basics we refer to [5].

Matrices that have the following structure, defined through a set of small matrices $\{A_k, B_k, C_k, D_k, A'_k, B'_k, C'_k\}$

$$\begin{pmatrix} D_1 & B_1C_2 & B_1A_2C_3 & \dots & B_1A_2\dots A_{n-1}C_n \\ B'_2C'_1 & D_2 & B_2C_3 & \dots & B_2A_3\dots A_{n-1}C_n \\ B'_3A'_2C'_1 & B'_3C'_2 & D_3 & \dots & B_3A_4\dots A_{n-1}C_n \\ & & \ddots & \ddots & \ddots \\ & & & B'_nC'_{n-1} & D_n \end{pmatrix}$$

are called matrices with sequentially semi-separable structure, and the sequence of matrices is called the state realization of the sequentially semi-separable structure of the matrix [5]. Let T be such a matrix, then the realization matrices correspond to a computational schema for the input-output product $y = uT$ involving a set of intermediate so-called state vectors $\{x_k, x'_k\}$ that are computed recursively

$$\begin{cases} x_{k+1} &= x_kA_k + u_kB_k \\ x'_{k-1} &= x'_kA'_k + u_kB'_k \\ y_k &= x_kC_k + x'_kC'_k + u_kD_k. \end{cases}$$

Rewritten in global operator form by assembling the matrices A_k, B_k etc... as diagonal operators on spaces of sequences of appropriate dimensions

$$A = \begin{bmatrix} \ddots & & 0 \\ & A_k & \\ 0 & & \ddots \end{bmatrix} B = \begin{bmatrix} \ddots & & 0 \\ & B_k & \\ 0 & & \ddots \end{bmatrix}$$

etc... and defining the shift-operator Z as $(uZ)_i = u_{i-1}$ we obtain a compact representation of T in terms of its structural matrices as

$$T = D + BZ(I - AZ)^{-1}C + B'Z^{-1}(I - A'Z^{-1})^{-1}C'.$$

Of course, all dimensions of matrices and vectors have to match wherever needed. The structural matrices are often brought together in view of this as

$$\mathbf{T}_c = \begin{bmatrix} A & C \\ B & D \end{bmatrix}, \ \mathbf{T}_a = \begin{bmatrix} A' & C' \\ B' & 0 \end{bmatrix}, \tag{11}$$

which is a 4×4 block matrix with diagonal entries. We now briefly discuss matrix operations using the semi separable structure. We first concentrate on upper triangular matrices for which the accented quantities are zero. Let us, for convenience, define the diagonal shift operator $T^{(1)}$ by

$$Z_M T^{(1)} = TZ_N$$

that is, $T^{(1)} = Z^{-1}TZ$, then $T^{(1)}$ is the operator T whose representation is shifted one position into the South-East direction: $(T^{(1)})_{i,j} = T_{i-1,j-1}$. More generally,

the kth diagonal shift of T into the southeast direction along the diagonals of T is defined by

$$T^{(k)} = (Z^k)^{-1} T Z^k.$$

Equivalently, $(T^{(k)})_{i,j} = T_{i-k,j-k}$.

4.1. State transformations

Two realizations $\{A_1, B_1, C_1, D\}$ and $\{A_2, B_2, C_2, D\}$ are called equivalent if their respective state vectors are related through an invertible transformation R. We have then

$$\begin{bmatrix} A_2 & C_2 \\ B_2 & D \end{bmatrix} = \begin{bmatrix} R & \\ & I \end{bmatrix} \begin{bmatrix} A_1 & C_1 \\ B_1 & D \end{bmatrix} \begin{bmatrix} [R^{(-1)}]^{-1} & \\ & I \end{bmatrix}$$
$$R^{(-1)} = ZRZ^{-1}.$$

We say that a realization is minimal if none of the dimensions of the state vectors can be reduced further. It is known [5] that these minimal dimensions form a unique sequence and that two minimal realizations are related through an invertible transformation matrix.

4.2. Sum of two realizations

let T_1, T_2 be two upper triangular matrices, with realizations A_1, B_1, C_1, D_1 and A_2, B_2, C_2, D_2, respectively. Then the sum of these two operators, $T = T_1 + T_2$, has a realization given directly in terms of these two realizations as

$$\begin{bmatrix} A & C \\ B & D \end{bmatrix} = \left[\begin{array}{cc|c} A_1 & 0 & C_1 \\ 0 & A_2 & C_2 \\ \hline B_1 & B_2 & D_1 + D_2 \end{array}\right].$$

The state dimension sequence of this realization is equal to the sum of the state dimension sequences of T_1 and T_2. It is, however, not necessarily minimal even if the component dimensions are.

4.3. Product of two realizations

The product of $T = T_1 T_2$ can also be obtained using realizations by

$$\begin{bmatrix} A & C \\ B & D \end{bmatrix} = \left[\begin{array}{cc|c} A_1 & C_1 B_2 & C_1 D_2 \\ 0 & A_2 & C_2 \\ \hline B_1 & D_1 B_2 & D_1 D_2 \end{array}\right].$$

In this case also the dimension of the given product realization is the sum of the dimensions of the components. It is not necessarily minimal even though the realizations of the factors are – there may be cancellations between the factors.

4.4. Realization of an upper inverse

Let T be an invertible upper triangular matrix, and suppose that it is known that T^{-1} is also upper, then the D matrix in the realization has to be square invertible

and a realization for T^{-1} is given by [5]

$$\begin{bmatrix} A' & C' \\ B' & D' \end{bmatrix} = \begin{bmatrix} A - CD^{-1}B & -CD^{-1} \\ D^{-1}B & D^{-1} \end{bmatrix}.$$

This realization for T^{-1} will be minimal if the realization for T is.

4.5. Cholesky factorization

We now return to the mixed upper-lower case. Given $T > 0$ and let the upper triangular part of T have a minimal state space realization A_k, B_k, C_k, D_k. Let $T = F^H F$ where F is upper triangular. The main property of relevance here is that the upper factor F has a minimal state space realization of the same dimensions as the upper part of T. It is given by
Then a realization $A_{F,k}, B_{F,k}, C_{F,k}, D_{F,k}$ of F is given by (superscript $\cdot^H$ indicates Hermitian conjugation)

$$\begin{cases} A_{F,k} = A_k \\ C_{F,k} = C_k \\ D_{F,k} = (D_k - C_k^H \Lambda_k C_k)^{-1/2} \\ B_{F,k} = D_{F,k}^{-1}(B_k - C_k^H \Lambda_k A_k) \end{cases}$$

where Λ_k is given by the recursion

$$\Lambda_{k+1} = A_k^H \Lambda_k A_k + B_{F,k}^H B_{F,k}.$$

T has to be positive definite for this recursion to work out. In case that turns out not to be so, then at a certain point k in the recursion $D_k - C_k^H \Lambda_k C_k$ will turn out to be non-positive definite, leading to a non positive square root.

5. Efficient Schur reduction: the semi-separable case

In this section we consider the rather more general case where the matrix to be reduced has the form

$$\Phi = \begin{pmatrix} M_1 & -L_1^H & & & \\ -L_1 & M_2 & -L_1^H & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & M_{m-1} & -L_{m-1}^H \\ & & & -L_{m-1} & M_m \end{pmatrix} \tag{12}$$

5.1. State space realizations of matrices M and L

Φ consists of block matrices M_k and L_k. Dropping the index k whenever clear from the context, we assume further that the matrices M and L have general

sequentially semi-separable realizations of the type

$$\begin{aligned} M_k &= \{A_{M_c}, A_{M_a}, B_{M_c}, B_{M_a}, C_{M_c}, C_{M_a}, D_M\} \\ &= D_M + B_{M_c} Z (I - A_{M_c} Z)^{-1} C_{M_c} \\ &\quad + B_{M_a} Z^{-1} (I - A_{M_a} Z^{-1})^{-1} C_{M_a} \\ L_k &= \{A_{L_c}, A_{L_a}, B_{L_c}, B_{L_a}, C_{L_c}, C_{L_a}, D_L\} \\ &= D_L + B_{L_c} Z (I - A_{L_c} Z)^{-1} C_{L_c} \\ &\quad + B_{L_a} Z^{-1} (I - A_{L_a} Z^{-1})^{-1} C_{L_a}. \end{aligned}$$

The realization of a sequentially semi-separable structure matrix can be computed by a low rank factorizations of some off-diagonal blocks called Hankel blocks, as described in [8]. In the tri-diagonal case the derivation is trivial.

5.2. Schur complements in the state space formalism

Because of the Hermitian structure of Φ, M and L, we may assume the realizations of M and L to have

$$\begin{aligned} A_{M_a} &= A_{M_c}^H, & & & A_{L_a} &= A_{L_c}^H, \\ B_{M_a} &= C_{M_c}^H, & \text{and} & & B_{L_a} &= C_{L_c}^H, \\ C_{M_a} &= B_{M_c}^H, & & & C_{L_a} &= B_{L_c}^H. \end{aligned}$$

From equation (10), each S_k will be Hermitian matrix as well, and hence can be Cholesky factorized as $S_k = F_k^H F_k$.

Assuming that the upper triangular matrix F_k has a realization

$$\{A_{Fc}, B_{Fc}, C_{Fc}, D_F\},$$

we now try to find the realization of upper triangular matrix F_{k+1} using only state space data.

$$\begin{aligned} S_{k+1} &= F_{k+1}^H F_{k+1} \\ &= M_{k+1} - L_k (F_k^H F_k)^{-1} L_k^H \\ &= M_{k+1} - L_k F_k^{-1} F_k^{-H} L_k^H. \end{aligned}$$

Let $E_k = F_k^{-1}$ and let us factorize L_k to be $L_k = L_{1_k} L_{2_k}$ and let the Cholesky factorization of M_k be $M_k = X_k X_k^H$, with E_k and L_{2_k} upper triangular matrices and L_{1_k} and X_k lower triangular. We get

$$F_{k+1}^H F_{k+1} = X_{k+1} X_{k+1}^H - L_{1_k} L_{2_k} E_k E_k^H L_{2_k}^H L_{1_k}^H$$

Let $G_k = L_{2_k} E_k$, which is an upper triangular matrix. To make the computation simple, we convert each G_k matrix to a lower triangular matrix H_k, where $G_k G_k^H = H_k H_k^H$. We get

$$\begin{aligned} F_{k+1}^H F_{k+1} &= X_{k+1} X_{k+1}^H - L_{1_k} G_k G_k^H L_{1_k}^H \\ &= X_{k+1} X_{k+1}^H - L_{1_k} H_k H_k^H L_{1_k}^H. \end{aligned}$$

Let $Y_k = L_{1_k} H_k$ (lower triangular), then

$$F_{k+1}^H F_{k+1} = X_{k+1} X_{k+1}^H - Y_k Y_k^H.$$

Dropping the index k in the following sections for simplification, and assuming the realization of any upper matrix O to be

$$\begin{aligned} O &= \{A_O, B_O, C_O, D_O\} \\ &= D_O + B_O Z (I - A_O Z)^{-1} C_O \end{aligned}$$

and the realization of any lower matrix P

$$\begin{aligned} P &= \{A_P, B_P, C_P, D_P\} \\ &= D_P + B_P Z^{-1} (I - A_P Z^{-1})^{-1} C_P. \end{aligned}$$

The following subsections explain how to get the state space of

$$F_{k+1} = \{A'_F, B'_F, C'_F, D'_F\} \quad \text{from} \quad F_k = \{A_F, B_F, C_F, D_F\}$$

using the basic steps explained above.

5.2.1. From F_{k-1} to E_k. As F_{k-1} is an upper triangular matrix, we can get the realization of $E_k = F_{k-1}^{-1}$ directly by time varying system theory as:

$$\begin{aligned} A_E &= A_F - C_F D_F^{-1} B_F \\ B_E &= D_F^{-1} B_F \\ C_E &= -C_F D_F^{-1} \\ D_E &= D_F^{-1}. \end{aligned}$$

5.2.2. Factorize L_k to L_{1_k} and L_{2_k}. where each L_{1_k} is an lower triangular matrix and L_{2_k} is a upper triangular matrix.

$$\begin{aligned} A_{l_1} &= A_{L_a}; \\ A_{l_2} &= A_{L_c}; \\ B_{l_1} &= B_{L_a}; \\ C_{l_2} &= C_{L_c}; \\ D_{l_2} &= I; \\ D_{l_1} &= (D_L - B_{L_a} \Lambda C_{L_c}) D_{l_2}^{-1} \\ B_{l_2} &= D_{l_1}^{-1} (B_{L_c} - B_{L_a} \Lambda A_{L_c}) \\ C_{l_1} &= (C_{L_a} - A_{L_a} \Lambda C_{L_c}) D_{l_2}^{-1} \\ \Lambda^{(1)} &= C_{l_1} B_{l_2} + A_{l_1} \Lambda A_{l_2} \end{aligned}$$

where $\Lambda^{(1)}$ stands for Λ_{k+1} and $\Lambda^{(-1)}$ stands for Λ_{k-1}.

5.2.3. $G_k = L_{2_k} E_k$**.** Both L_{2_k} and E_k are upper triangular matrices, so does G_k.

$$\begin{aligned} A_G &= \begin{bmatrix} A_{l_2} & C_{l_2} B_E \\ 0 & A_E \end{bmatrix} \\ B_G &= \begin{bmatrix} B_{l_2} & D_{l_2} B_E \end{bmatrix} \\ C_G &= \begin{bmatrix} C_{l_2} D_E \\ C_E \end{bmatrix} \\ D_G &= D_{l_2} D_E. \end{aligned}$$

5.2.4. From G_k **to** H_k**.** G_k are upper triangular matrices while H_k are lower triangular matrices with $G_k G_k^H = H_k H_k^H$.

$$\begin{aligned} A_H &= A_G^H; \\ B_H &= D_G C_G^H + B_G \Lambda A_G^H; \\ D_H &= (D_G D_G^H + B_G \Lambda B_G^H - B_H \Gamma B_H^H)^{1/2} \\ C_H &= (B_G^H - A_G^H \Gamma B_H^H) D_H^{-1} \end{aligned}$$

where

$$\begin{aligned} \Lambda^{(-1)} &= C_G C_G^H + A_G \Lambda A_G^H \\ \Gamma^{(1)} &= C_H C_H^H + A_H \Gamma A_H^H. \end{aligned}$$

5.2.5. $Y_k = L_{1_k} H_k$**.** Y_k, L_{1_k}, and H_k are all lower triangular matrices,

$$\begin{aligned} A_Y &= \begin{bmatrix} A_H & 0 \\ C_{l_1} B_H & A_{l_1} \end{bmatrix} \\ B_Y &= \begin{bmatrix} D_{l_1} B_H & B_{l_1} \end{bmatrix} \\ C_Y &= \begin{bmatrix} C_H \\ C_{l_1} D_H \end{bmatrix} \\ D_Y &= D_{l_1} D_H. \end{aligned}$$

5.2.6. Factorize $M_k = X_k X_k^H$**.** We can get lower triangular matrix X_k from M_k by Cholesky factorization again.

$$\begin{aligned} A_X &= A_M^H \\ B_X &= C_M^H \\ D_X &= (D_M - C_M^H \Gamma C_M)^{1/2} \\ C_X &= (B_M^H - A_M^H \Gamma C_M) D_X^{-1} \end{aligned}$$

where

$$\Gamma^{(1)} = C_X C_X^H + A_M^H \Gamma A_M.$$

5.2.7. Get F_{k+1} from $F_{k+1}^H F_{k+1} = X_{k+1} X_{k+1}^H - Y_k Y_k^H$. Compute lower triangular matrix $W_k = X_k^{-1} Y_k$,

$$\begin{aligned}
A_{X^{-1}} &= A_X - C_X D_X^{-1} B_X \\
B_{X^{-1}} &= -D_X^{-1} B_X \\
C_{X^{-1}} &= C_X D_X^{-1} \\
D_{X^{-1}} &= D_X^{-1}
\end{aligned}$$

$$\begin{aligned}
A_W &= \begin{bmatrix} A_Y & 0 \\ C_{X^{-1}} B_Y & A_{X^{-1}} \end{bmatrix} \\
B_W &= \begin{bmatrix} D_{X^{-1}} B_Y & B_{X^{-1}} \end{bmatrix} \\
C_W &= \begin{bmatrix} C_Y \\ C_{X^{-1}} D_Y \end{bmatrix} \\
D_W &= D_{X^{-1}} D_Y .
\end{aligned}$$

Let lower triangular matrix Δ satisfy $\Delta\Delta^H = I - WW^H$, and after Cholesky factorization, we obtain

$$\begin{aligned}
A_\Delta &= A_W \\
B_\Delta &= -B_W \\
D_\Delta &= I - D_W D_W^H - B_W \Lambda B_W^H - B_W \Gamma B_W^H)^{1/2} \\
C_\Delta &= (C_W D_W^H + A_W \Lambda B_W^H + A_W \Gamma B_W^H) D_\Delta^{-1}
\end{aligned}$$

where

$$\begin{aligned}
\Lambda^{(1)} &= C_W C_W^H + A_W \Lambda A_W^H \\
\Gamma^{(1)} &= C_\Delta C_\Delta^H + A_W \Gamma A_W^H .
\end{aligned}$$

Multiply $F = X\Delta$. We finally get the lower triangular matrix F'_{k+1} and upper triangular matrix $F_{k+1} = F'_{k+1}$ as

$$\begin{aligned}
A_F &= \begin{bmatrix} A_\Delta^H & B_\Delta^H C_X^H \\ 0 & A_X^H \end{bmatrix} \\
B_F &= \begin{bmatrix} C_\Delta^H & D_\Delta^H C_X^H \end{bmatrix} \\
C_F &= \begin{bmatrix} B_\Delta^H D_X^H \\ B_X^H \end{bmatrix} \\
D_F &= D_\Delta^H D_X^H .
\end{aligned}$$

5.3. Model reduction

The resulting Schur complement factor F_k at each step in the recursion can be model reduced by reduction on both reachability Gramian and observability Gramian in the state space. Let us assume that the realization of F_k obtained in the recursion step is $\{A, B, C, D\}$, and do the model reduction on reachability

Gramian Λ_c first, which results the reduced realization $\{A'_{11}, B'_1, C'_1, D'\}$. Then we reduce it again on observability Gramian Λ_o on $\{A'_{11}, B'_1, C'_1, D'\}$, resulting the final reduced realization $\{A''_{11}, B''_1, C''_1, D''\}$. More details can be found in [2].

5.3.1. On reachability Gramians. Let Λ_c be the reachability Gramian of the given realization $\{A, B, C, D\}$ of T, then it has the eigenvalue decomposition:

$$\Lambda_c = R_c^H \begin{bmatrix} \widetilde{\Lambda}_c & 0 \\ 0 & 0 \end{bmatrix} R_c, \quad R_c = \begin{bmatrix} \widetilde{R}_c \\ * \end{bmatrix},$$

where $\widetilde{\Lambda}_c$ is a diagonal matrix containing the nonzero eigenvalues of Λ_c, while R_c can be chosen unitary, and $\widetilde{R}_c$ contains the columns of R_c corresponding to the entries in $\widetilde{\Lambda}_c$. Apply R_c as a state transformation to T, we get an equivalent realization $T' = \{A', B', C', D'\}$,

$$\begin{aligned} \begin{bmatrix} A' & C' \\ B' & D' \end{bmatrix} &= \begin{bmatrix} R_c & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} A & C \\ B & D \end{bmatrix} \begin{bmatrix} R_c^{(-1)H} & 0 \\ 0 & I \end{bmatrix} \\ &= \begin{bmatrix} R_c A R_c^{(-1)H} & R_c C \\ B R_c^{(-1)H} & D \end{bmatrix} \\ \Lambda'_c &= \begin{bmatrix} \widetilde{\Lambda}_c & 0 \\ 0 & 0 \end{bmatrix} \end{aligned}$$

where Λ'_c is the reachability Gramian of T', and satisfies the Lyapunov equation $\Lambda_c'^{(-1)} = A'^H \Lambda'_c A' + B'^H B'$. Partition A', B', and C' according to that of R_c,

$$\begin{aligned} A' &= \begin{bmatrix} A'_{11} & 0 \\ A'_{21} & A'_{22} \end{bmatrix}, \quad C' = \begin{bmatrix} C'_1 \\ C'_2 \end{bmatrix}, \\ B' &= \begin{bmatrix} B'_1 & 0 \end{bmatrix}, \qquad D' = D. \end{aligned}$$

Because $B_2'^H B'_2 + A_{12}'^H \Lambda_{11} A'_{12} = 0$ and $\Lambda_{11} > 0$, we have $B_2 = A_{12} = 0$.

Then the model reduced realization is

$$\begin{bmatrix} A'_{11} & C'_1 \\ B'_1 & D' \end{bmatrix} = \begin{bmatrix} \widetilde{R}_c A \widetilde{R}_c^{(-1)H} & \widetilde{R}_c C \\ B \widetilde{R}_c^{(-1)H} & D \end{bmatrix}.$$

5.3.2. On observability Gramians. Similarly, the observability Gramian Λ_o of realization $\{A'_{11}, B'_1, C'_1, D'\}$ can be decomposed as:

$$\Lambda_o = R_o^{-1} \begin{bmatrix} \widetilde{\Lambda}_o & 0 \\ 0 & 0 \end{bmatrix} R_o^{-H}, \quad R_o = \begin{bmatrix} \widetilde{R}_o & * \end{bmatrix},$$

Apply the unitary matrix R_o to the realization $\{A'_{11}, B'_1, C'_1, D'\}$, we get realization,

$$\begin{aligned}
\begin{bmatrix} A'' & C'' \\ B'' & D'' \end{bmatrix} &= \begin{bmatrix} R_o & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} A'_{11} & C'_1 \\ B'_1 & D' \end{bmatrix} \begin{bmatrix} R_o^{(-1)H} & 0 \\ 0 & I \end{bmatrix} \\
&= \begin{bmatrix} R_o A'_{11} R_o^{(-1)H} & R_o C'_1 \\ B'_1 R_o^{(-1)H} & D' \end{bmatrix} \\
\Lambda''_o &= \begin{bmatrix} \widetilde{\Lambda}_o & 0 \\ 0 & 0 \end{bmatrix}
\end{aligned}$$

and

$$\begin{aligned}
A'' &= \begin{bmatrix} A''_{11} & A''_{12} \\ 0 & A''_{22} \end{bmatrix}, \quad C'' = \begin{bmatrix} C''_1 \\ 0 \end{bmatrix}, \\
B'' &= \begin{bmatrix} B''_1 & B''_2 \end{bmatrix}, \quad D'' = D.
\end{aligned}$$

So the final reduced realization is

$$\begin{bmatrix} A''_{11} & C''_1 \\ B''_1 & D'' \end{bmatrix} = \begin{bmatrix} \widetilde{R}_o A''_{11} \widetilde{R}_o^{(-1)H} & \widetilde{R}_o C''_1 \\ B''_1 \widetilde{R}_o^{(-1)H} & D'' \end{bmatrix}.$$

6. Exact solutions for the 3D model cases

In this section we present the exact solutions (in closed form) for the two regular 3D-cases announced earlier. The method we use extends the method presented in [4, 7] to the 3D case. In particular, we compute the exact value of the Schur complements and their limiting fixed point. In the next section we shall use these results to evaluate pre-conditioners in a number of situations.

Reverting back to the notation of Section 2 we have that the sequence of Schur complements from the top-left subblock to the down-right subblock is given by equation (10).

Because of the positive definiteness of the Poisson matrix, all subsequent Schur complements are positive definite as well, and the recursion will converge, as we shall see, to a fixed point matrix S_∞, which we shall evaluate. It turns out that the latter is actually a very good approximant of the actual Schur complements for larger values of k, the convergence being pretty fast.

6.1. Block symmetric tri-diagonal Toeplitz matrix

In this section we consider only real symmetric (block-) tri-diagonal Toeplitz matrices, in which the blocks themselves are real symmetric (block) tri-diagonal and

Toeplitz. For brevity, we introduce the shorthand tridiag(A, B) for the matrix

$$\text{tridiag}(A,B) = \begin{pmatrix} A & B^H & & & \\ B & A & B^H & & \\ & B & A & \ddots & \\ & & \ddots & \ddots & B \\ & & & B & A \end{pmatrix}. \tag{13}$$

In this hierarchy we indicate the depth d by a super-index, and the corresponding size of the matrices by n_d. At level 0 we then have

$$A^{(0)} = \text{tridiag}(A^{(1)}, B^{(1)}). \tag{14}$$

The hierarchy terminates when a certain depth $d = D - 1$ is reached, where each of the sub-blocks is a symmetric tri-diagonal Toeplitz matrix having the form

$$A^{(D-1)} = \text{tridiag}(a, b) \tag{15}$$

with scalar entries.

Theorem 1. *Given a block real symmetric tri-diagonal Toeplitz matrices $T = tridiag(A, B)$ at a hierarchical depth d with n_d number of sub-blocks on its diagonal, and let the $d-1$ depth sub-block matrices A and B have eigenvalues $\Lambda_A = \text{diag}(\lambda_{A;1}, \ldots, \lambda_{A;p})$ and $\Lambda_B = \text{diag}(\lambda_{B;1}, \ldots, \lambda_{B;p})$ respectively, with dimension $p \times p$ and the same normalized eigenvector matrix U, then the matrix T has eigenvalues $\Lambda_T = \Lambda_A + 2\Lambda_B \cos\frac{k\pi}{n_d+1}$, where $1 \le k \le n_d$ with corresponding normalized block eigenvectors given by*

$$V_k = \begin{pmatrix} U\sin(\frac{1k\pi}{n_d+1})/\sigma \\ U\sin(\frac{2k\pi}{n_d+1})/\sigma \\ \vdots \\ U\sin(\frac{n_dk\pi}{n_d+1})/\sigma \end{pmatrix} \qquad \textit{where} \qquad \sigma = \sqrt{\sum_{i=1}^{n_d} \sin^2\left(\frac{ik\pi}{n_d+1}\right)}.$$

The theorem generalizes a result of [6] about tri-diagonal Toeplitz matrices reproduced hereunder.

Theorem 2. *If C is an $n \times n$ tri-diagonal Toeplitz matrix with*

$$C = \begin{pmatrix} a & b & & & \\ c & a & b & & \\ & c & a & \ddots & \\ & & \ddots & \ddots & b \\ & & & c & a \end{pmatrix} \tag{16}$$

then the eigenvalues of C are given by

$$\lambda_j = a + 2b\sqrt{\frac{c}{b}}\cos(\frac{j\pi}{n+1}), \tag{17}$$

where $1 \leq j \leq n$, *and corresponding eigenvectors are given by*

$$x_j = \begin{pmatrix} (\frac{c}{b})^{1/2} \sin(\frac{1j\pi}{n+1}) \\ (\frac{c}{b})^{2/2} \sin(\frac{2j\pi}{n+1}) \\ (\frac{c}{b})^{3/2} \sin(\frac{3j\pi}{n+1}) \\ \vdots \\ (\frac{c}{b})^{n/2} \sin(\frac{nj\pi}{n+1}) \end{pmatrix}. \tag{18}$$

Proof of Theorem 1. The theorem follows from the fact that all matrices involved commute. This is true at the bottom level $D-1$ of the hierarchy since at that level the off-diagonal entries of the matrices are equal resulting in eigenvectors that can be chosen equal. Moving up the hierarchy, the constitutive block-matrices keep commuting. We may assume recursively that A and B are diagonalized by the same eigenvector matrix U,

$$A = U\Lambda_A U^H$$
$$B = U\Lambda_B U^H$$

then

$$T = \begin{pmatrix} U & & & \\ & U & & \\ & & \ddots & \\ & & & U \end{pmatrix} \begin{pmatrix} \Lambda_A & \Lambda_B & & \\ \Lambda_B & \Lambda_A & & \\ & & \ddots & \Lambda_B \\ & & \Lambda_B & \Lambda_A \end{pmatrix} \begin{pmatrix} U^H & & & \\ & U^H & & \\ & & \ddots & \\ & & & U^H \end{pmatrix}.$$

Let us denote

$$T' = \begin{pmatrix} \Lambda_A & \Lambda_B & & \\ \Lambda_B & \Lambda_A & & \\ & & \ddots & \Lambda_B \\ & & \Lambda_B & \Lambda_A \end{pmatrix} \tag{19}$$

and apply Theorem 2 on T', (notice that after proper permutation P, $T" = PT'P$ is block diagonal matrix with each sub-block a symmetric tri-diagonal Toeplitz matrix,) we get the eigenvalues of T':

$$\Lambda_{T'_k} = \Lambda_A + 2\Lambda_B \cos\left(\frac{k\pi}{n_d + 1}\right), \tag{20}$$

where $1 \leq k \leq n_d$, and the corresponding eigenvector block is

$$V_j' = \begin{pmatrix} I_{p\times p} \sin(\frac{1k\pi}{n_d+1})/\sigma \\ I_{p\times p} \sin(\frac{2k\pi}{n_d+1})/\sigma \\ I_{p\times p} \sin(\frac{3k\pi}{n_d+1})/\sigma \\ \vdots \\ I_{p\times p} \sin(\frac{nk\pi}{n_d+1})/\sigma \end{pmatrix}. \tag{21}$$

where

$$\sigma = \sqrt{\sum_{i=1}^{n_d} \sin^2\left(\frac{ik\pi}{n_d+1}\right)}$$

$$T = \begin{pmatrix} U & & & \\ & U & & \\ & & \ddots & \\ & & & U \end{pmatrix} V' \Lambda_{T'} V'^H \begin{pmatrix} U^H & & & \\ & U^H & & \\ & & \ddots & \\ & & & U^H \end{pmatrix}.$$

Therefore, the eigenvalues and eigenvectors of T are: $\Lambda_T = \Lambda_A + 2\Lambda_B \cos\frac{k\pi}{n_d+1}$, where $1 \le k \le n_d$. The corresponding block eigenvector is

$$V_k = \begin{pmatrix} U\sin(\frac{1k\pi}{n_d+1})/\sigma \\ U\sin(\frac{2k\pi}{n_d+1})/\sigma \\ \vdots \\ U\sin(\frac{n_dk\pi}{n_d+1})/\sigma \end{pmatrix}.$$

The block eigen-structure can hence be applied hierarchically in this case leading to a diagonalized matrix at the top level of the hierarchy. From this it follows that subsequent Schur complements and the fixed point solution of the Schur-complement equation can be found explicitly. On the diagonalized matrix all the Schur complements are diagonal as well, as well as the fixed point solution. This is stated in the following (almost obvious) theorem.

Theorem 3. *Let Φ be given by* (2)*, and assume that for any block-dimension n, Φ is strictly positive definite. Then $\Lambda_M > 2\Lambda_L$ and the fixed point solution for the Schur complement S_∞ is given by*

$$S_\infty = \frac{1}{2} V \left(\Lambda_M + \sqrt{\Lambda_M^2 - 4\Lambda_L^2}\right) V^H. \tag{22}$$

where

$$\Lambda_{M_j} = \Lambda_O + 2\Lambda_P \cos\left(\frac{j\pi}{n+1}\right),$$

$$\Lambda_{L_j} = \Lambda_R + 2\Lambda_Q \cos\left(\frac{j\pi}{n+1}\right),$$

$$\lambda_{O_i} = a_O + 2b_O \cos\left(\frac{i\pi}{n+1}\right),$$

$$\lambda_{P_i} = a_P + 2b_P \cos\left(\frac{i\pi}{n+1}\right),$$

$$\lambda_{Q_i} = a_Q + 2b_Q \cos\left(\frac{i\pi}{n+1}\right),$$

$$\lambda_{R_i} = a_R + 2b_R \cos\left(\frac{i\pi}{n+1}\right),$$

$$v_j = \frac{1}{\sqrt{\sum_{k=1}^{n} \sin^2(\frac{kj\pi}{n+1})}} \begin{pmatrix} U\sin(\frac{1j\pi}{n+1}) \\ U\sin(\frac{2j\pi}{n+1}) \\ \vdots \\ U\sin(\frac{nj\pi}{n+1}) \end{pmatrix}.$$

$$u_i = \frac{1}{\sqrt{\sum_{k=1}^{n} \sin^2(\frac{ki\pi}{n+1})}} \begin{pmatrix} \sin(\frac{1i\pi}{n+1}) \\ \sin(\frac{2i\pi}{n+1}) \\ \vdots \\ \sin(\frac{ni\pi}{n+1}) \end{pmatrix}$$

and $1 \le j \le n, 1 \le i \le n$.

Proof. In this case the hierarchical decomposition is just two levels deep (corresponding to the 3D case) and the matrices involved can easily be written down explicitly. It is easy to tell from the theorem that M and L share the same eigenvector matrix V and $M = V\Lambda_M V^H$, $L = V\Lambda_L V^H$. Hence we have $S_0 = U\Lambda_0 U^H$ and $S_k = U\Lambda_k U^H$ for the same collection of normalized eigenvectors assembled in U. The Schur recursion then becomes a collection of scalar recursions given by

$$\begin{cases} \Lambda_0 = \Lambda_M, \\ \Lambda_{k+1} = \Lambda_M - \Lambda_L \Lambda_k^{-1} \Lambda_L^H \ (k = 0, 1, \dots). \end{cases}$$

The intermediate matrices at the hierarchical level 2 have a diagonalization represented by $\Phi' = \text{tridiag}(\Lambda_M, \Lambda_L)$ which transforms, by reordering of rows and columns into a direct sum of Toeplitz matrices of the form $\text{tridiag}(m_k, \ell_k)$. Since all these are strictly positive definite for any dimension by assumption, it must be that $m_k > 2\ell_k$, by a well-known property of Toeplitz matrices (corresponding to the fact that the limiting spectrum must be (strictly) positive definite – see [6]). The scalar iteration

$$x_{k+1} = m - \ell x_k^{-1} \ell \quad \text{with} \quad m > 2\ell > 0$$

converges to

$$x = \frac{1}{2}(m + \sqrt{m^2 - 4l^2}).$$

Since $\Lambda_M > 2\Lambda_L$, the iteration converges for each entry in the recursion, leading to the result claimed in the theorem. □

6.2. The block-circulant case

A theory similar to the block triangular Toeplitz case can be set up for (block-) circulant matrices. It covers the case of a regular grid on which periodic boundary conditions are in effect. The eigenvalue analysis in this case leads to Fourier transformation. We briefly summarize the results.

A block circulant matrix is completely determined by its first block row. This we denote as follows

$$\mathrm{circ}(A_k : k = 0 \cdots n-1) = \begin{pmatrix} A_0 & A_1 & A_2 & \dots & A_{n-1} \\ A_{n-1} & A_0 & A_1 & \dots & \vdots \\ A_{n-2} & A_{n-1} & A_0 & \dots & \vdots \\ \vdots & \dots & \dots & \ddots & A_1 \\ A_1 & A_2 & \dots & A_{n-1} & A_0 \end{pmatrix}.$$

We consider a hierarchy of such matrices, meaning that at each level we dispose of a circulant block matrix with blocks that are themselves circulant block matrices up to a level where the components are just scalar circulant matrices – this corresponds to a physical situation in which periodic boundary conditions are in effect in every dimension. We shall indicate the hierarchical level with a super-index. Assuming the overall hierarchical depth is D, $A^{(0)}$ indicates the top matrix in the hierarchy. Then

$$A^{(0)} = \mathrm{circ}(A_k^{(1)} : k = 1 \cdots n_1)$$

in which

$$A_k^{(1)} = \mathrm{circ}(A_{k;\ell}^{(2)} : \ell = 1 \cdots n_2)$$

etc... This keeps going until depth $d = D - 1$, where each of the sub-blocks is a regular circulant matrix with scalar entries.

Theorem 4. *Given a one depth block circulant matrix with $n \times n$ sub-blocks*

$$A = \begin{pmatrix} A_0 & A_1 & A_2 & \dots & A_{n-1} \\ A_{n-1} & A_0 & A_1 & \dots & \vdots \\ A_{n-2} & A_{n-1} & A_0 & \dots & \vdots \\ \vdots & \dots & \dots & \ddots & A_1 \\ A_1 & A_2 & \dots & A_{n-1} & A_0 \end{pmatrix}.$$

Assume that every sub-block A_k, $k = 1, \dots, n$ has the same dimension $p \times p$ and the same complete set of orthonormal eigenvectors assembled in the matrix U. Let the eigenvalue matrix of sub-block matrix A_k be $\Gamma_k = \mathrm{diag}(\gamma_{k;1}, \dots, \gamma_{k;p})$. Then the matrix A has eigenvalue $\Lambda_m = \mathrm{diag}(\lambda_{m;1}, \dots, \lambda_{m;p})$, with corresponding eigenvector block $x^{(m)}$, where

$$\lambda_{m;i} = \sum_{k=0}^{n-1} \gamma_{k;i} e^{\frac{-2\pi j m k}{n}},$$

$$x^{(m)} = \frac{1}{\sqrt{n}} [U, Ue^{\frac{-2\pi j m k}{n}}, \dots, Ue^{\frac{-2\pi j (n-1) k}{n}}]^T.$$

Proof. The equation $Ax = \lambda x$ for the eigenvalues and eigenvectors of A specializes to

$$\begin{pmatrix} A_0 & A_1 & \dots & A_{n-1} \\ A_{n-1} & A_0 & \dots & \vdots \\ A_{n-2} & A_{n-1} & \dots & \vdots \\ \vdots & \dots & \ddots & A_1 \\ A_1 & \dots & A_{n-1} & A_0 \end{pmatrix} \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_n \end{pmatrix} = \begin{pmatrix} \Lambda_1 & & \\ & \ddots & \\ & & \Lambda_n \end{pmatrix} \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_n \end{pmatrix} \tag{23}$$

where

$$\vec{x}_m = \begin{pmatrix} x_{m;1} \\ x_{m;2} \\ \vdots \\ x_{m;p} \end{pmatrix} \tag{24}$$

$$\Lambda_m = \begin{pmatrix} \lambda_{m;1} & & \\ & \ddots & \\ & & \lambda_{m;p} \end{pmatrix} \tag{25}$$

$$\sum_{k=0}^{n-1-m} A_k \vec{x}_{k+m} + \sum_{k=n-m}^{n-1} A_k \vec{x}_{k-n+m} = \vec{x}_m \Lambda_m \tag{26}$$

for $m = 1, \dots, n$.

Because

$$A_k = U \Gamma_k U^*. \tag{27}$$

Let

$$\vec{y}_m = U^* \vec{x}_m = \begin{pmatrix} y_{m;1} \\ y_{m;2} \\ \vdots \\ y_{m;p} \end{pmatrix} \tag{28}$$

then

$$\vec{x}_m = U \vec{y}_m, \tag{29}$$

$$\sum_{k=0}^{n-1-m} \Gamma_k \vec{y}_{k+m} + \sum_{k=n-m}^{n-1} \Gamma_k \vec{y}_{k-n+m} = \vec{y}_m \Lambda_m, \tag{30}$$

$$\sum_{k=0}^{n-1-m} \gamma_{k;i} y_{k+m;i} + \sum_{k=n-m}^{n-1} \gamma_{k;i} y_{k-n+m;i} = y_{m;i} \lambda_{m;i}, \tag{31}$$

where $m = 1, \dots, n$ and $i = 1, \dots, p$.

As this system of equations involves a scalar circulant matrix, namely

$$\operatorname{circ}(\gamma_{k;i} : k = 0 \cdots n-1),$$

the solution can be written down directly (the discrete Fourier transform of order n diagonalizes the matrix)

$$y_{k;i} = \rho^k, \tag{32}$$

with $\rho = e^{-\frac{2m\pi j}{n}}$. Furthermore

$$\lambda_{m;i} = \sum_{k=0}^{n-1} \gamma_{k;i} e^{\frac{-2\pi jmk}{n}}; \tag{33}$$

$$y_i = \frac{1}{\sqrt{n}}[1, e^{\frac{-2\pi jm}{n}}, \ldots, e^{\frac{-2\pi jm(n-1)}{n}}]^T. \tag{34}$$

Thus, for the sub-blocks,

$$\Lambda_m = \begin{pmatrix} \lambda_{m;1} & & \\ & \ddots & \\ & & \lambda_{m;p} \end{pmatrix}; \tag{35}$$

$$x^{(m)} = \begin{pmatrix} Uy_{i,1} \\ Uy_{i,2} \\ \vdots \\ Uy_{i,n} \end{pmatrix} \tag{36}$$

where I is the $p \times p$ identity matrix.

Therefore,

$$\Lambda_m = \begin{pmatrix} \lambda_{m;1} & & \\ & \ddots & \\ & & \lambda_{m;p} \end{pmatrix}, \tag{37}$$

$$\lambda_{m;i} = \sum_{k=0}^{n-1} \gamma_{k;i} e^{\frac{-2\pi jmk}{n}}, \tag{38}$$

$$x^{(m)} = \frac{1}{\sqrt{n}}[U^T, U^T e^{\frac{-2\pi jmk}{n}}, \ldots, U^T e^{\frac{-2\pi jm(n-1)k}{n}}]^T. \tag{39}$$

□

6.2.1. Application to the 3D Poisson equation on a regular grid. Using the notation of Section 2 we take the circulant version of the matrices $\Phi, M, L, \ldots$, i.e., the case $\epsilon = 1$. Application of Theorem 4 immediately provides the eigenvalues as samples of the Fourier transforms (it is just as convenient to handle the transforms directly):

$$\mathcal{F}(O) = -\frac{64}{15} + \frac{14}{15}\cos\theta$$
$$\mathcal{F}(P) = \frac{7}{15} + \frac{1}{5}\cos\theta$$
$$\mathcal{F}(R) = \frac{7}{15} + \frac{1}{5}\cos\theta$$

$$\begin{aligned}
\mathcal{F}(Q) &= \frac{1}{10} + \frac{1}{15}\cos\theta \\
\mathcal{F}(M) &= \mathcal{F}(O) + 2\mathcal{F}(P)\cos n\theta \\
&= -\frac{64}{15} + \frac{14}{15}\cos\theta + 2\left(\frac{7}{15} + \frac{1}{5}\cos\theta\right)\cos n\theta \\
\mathcal{F}(L) &= \mathcal{F}(R) + 2\mathcal{F}(Q)\cos n\theta \\
&= \frac{7}{15} + \frac{1}{5}\cos\theta + 2\left(\frac{1}{10} + \frac{1}{15}\cos\theta\right)\cos n\theta.
\end{aligned}$$

As in the previous schema, we can approximate the Schur complement sequence by calculating S_∞,

$$S_\infty = \frac{M + \sqrt{M^2 - 4LL^H}}{2} \tag{40}$$

with Fourier transform of S_∞

$$\begin{aligned}
\mathcal{F}(S_\infty) &= \mathcal{F}\left(\frac{M + \sqrt{M^2 - 4LL^H}}{2}\right) \qquad (41)\\
&= \frac{1}{2}(\mathcal{F}(M) + \sqrt{\mathcal{F}^2(M) - 4\mathcal{F}^2(L)}) \\
&= \frac{1}{15}[-32 + 7\cos\theta + 7\cos n\theta + 3\cos\theta\cos n\theta \\
&\quad + [5(-5 + 2\cos\theta + 2\cos n\theta + \cos\theta\cos n\theta) \\
&\qquad \cdots (-39 + 4\cos\theta + 4\cos n\theta + \cos\theta\cos n\theta)]^{\frac{1}{2}}].
\end{aligned}$$

This explicit representation of the fixed point for the Schur complement recursion, although interesting in its own right, allows us to evaluate various approximation schemas that lead to candidate pre-conditioners. We report on a number of results in the next section. Looking at the spectrum (formula 41) it is obvious that it oscillates strongly with a period π/m – in contrast to the 2D-case where the spectrum is very smooth with just a discontinuity in the first derivative at zero frequency [4]. As a consequence, any reasonable rational (finite state space) approximation of such a spectrum will need an order that is at least m. This shows that the 3D case is essentially different from 2D, where low order semi-separable approximation yield very good results. Hence, a different pre-conditioning strategy will have to be used in 3D and higher dimensions. The following section presents a preliminary investigation of this issue.

7. Decoupling strategies for matrices with 3D sparsity patterns

The goal of this section is to obtain insight in the possibilities of constructing pre-conditioners through decoupling, whereby the Schur complements that have to be introduced are being approximated by simpler matrices. This can happen at various positions in the schema, as we shall indicate further on in this section.

Since we dispose of exact inverses for special types of matrices (Poisson's equation on a regular grid) as presented in the previous sections, we can evaluate the various strategies on this very well-conditioned example. The idea behind this approach is that a pre-conditioning strategy should at least work well on simple straight examples, for there would be no hope for it to work in more complex cases if it already breaks down on the most well-conditioned realistic examples. We first describe the general strategies and then apply them on the Poisson case. The chapter has a very preliminary nature, as many possibilities have yet to be investigated.

As explained in the section "Hierarchical Decoupling Structure", the matrix on one layer of the symbolic hierarchy can be decoupled into several levels (all within that layer):

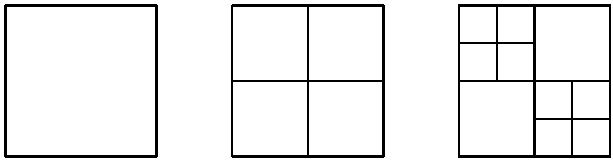

FIGURE 2. Level decomposition: level 0, level 1 and level 2 are shown

To keep terminology consistent we use the term 'layer' systematically to indicate the position in the symbolic hierarchy, while the term 'level' will be used for subdivisions within one layer as indicated by the diagrams above.

Positioning ourselves at a given layer of the hierarchy (say the top layer), if it consists of $n \times n$ sub-blocks from its next layer, then we dispose of $\log_2 n$ levels for decoupling totally at the given layer. Consider a point $k+1$ where an approximation of a Schur inverse has to be introduced. The exact solution is of course given by equation (10). The simplest approximant would of course be $S_{k+1} = M$, which would amount to putting $L = 0$. This we would refer to as a 'Jacobi step'. Notice that doing so, the resulting decoupled matrices remain positive definite (they are in fact more positive than the exact ones, as the difference is a semi-positive operator). This amount to a 'zero'th order' approximation. One step more (first order) would take $S_{k+1} = M - LM^{-1}L^H$, which again would produce a positive definite decoupling as can be seen from considering the positive definite block submatrix with rows and columns indexed by $k, k+1, \ldots$. This we would call a 'first-order approximant'. This process can of course be continued leading to second, third etc... order approximants. We show later that for the example the approximation error decreases quickly when higher orders are involved - the approximation error at the top layer of the hierarchy will appear to be the most significant factor.

The first-order approximation involves the inverse M^{-1}. This is a matrix at the next layer of the hierarchy. We may denote it as $S^{-1}_{k+1;-1}$ as we regress one stage in the Schur recursion. Similarly, the second order will involve a matrix $S_{k+1;-2}$, which would be obtained by regressing two stages – we have not studied

this case yet. As these inverses involve a lower layer of the hierarchy, again a Schur approximation can be introduced, but now involving matrices with a lower hierarchical structure. In the case of hierarchical depth $D = 3$, this may already involve a semi-separable approximation – we do present numerical results for this case where we restrict ourselves to a one-stage recursion (i.e., we approximate the relevant S_{k+1} as $M - LM^{-1}L^H$).

Matrix Φ is then first decoupled in the middle, where $k = \lfloor \frac{n}{2} \rfloor$. The process can then be repeated at the next levels on the same layer. We specify a level number $lvl1$ and decouple matrix Φ into 2^{lvl1} matrices as in the picture above, and approximate the Schur complement by $S_{k+1} = M - LM^{-1}L^H$ at the decoupling points while performing the exact recursion in between:

$$\begin{cases} S_0 = M, \\ S_{k+1} = M - LM^{-1}L^H, & k = \lfloor \frac{n}{2^{lvl1}} \rfloor, \lfloor \frac{2n}{2^{lvl1}} \rfloor, \dots \\ S_{k+1} = M - LS_k^{-1}L^H, & k = \text{else}. \end{cases} \tag{42}$$

Let us now specialize these ideas to our prototype example, the matrix representing the 3D regular Poisson problem with discretization based on the 27 points stencil. It has the form given in Fig. 3 (here specialized to $n = 8$). Each block in the matrix

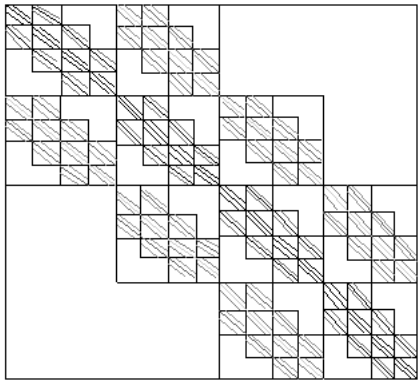

FIGURE 3. Structure of matrix Φ

represents a 2D plane, while each row corresponds to an array of points in the x-direction. The interactions in the y-direction take place within the blocks while the interactions in the z-direction create entries between blocks in the top layer. These interactions are shown in Fig. 4. where each of the xy-plane corresponds to a diagonal 2D block, each x line corresponds to a diagonal 1D block, and the lines between these points correspond to the off-diagonal blocks or entries, representing their relations. Introducing a cut in the top layer matrix (e.g., in the middle as shown in Fig. 5) forces the elimination of the cross dependencies between the layers, which are then incorporated as a (Schur-) correction $(-LS_k^{-1}L^H)$ on

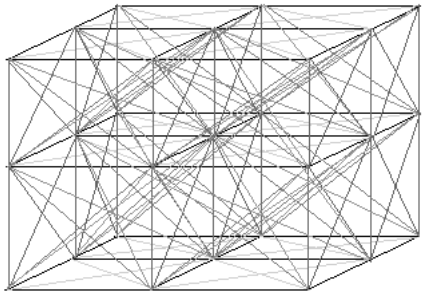

FIGURE 4. Physical model of matrix Φ

the next diagonal block. Our strategy hence consists in estimating rather than calculating that correction.

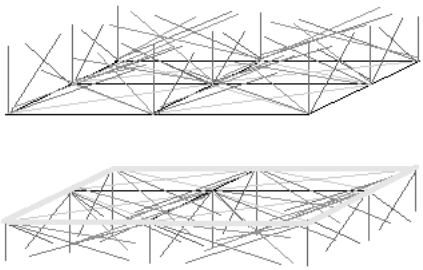

FIGURE 5. Cut on 3D layer (0 order)

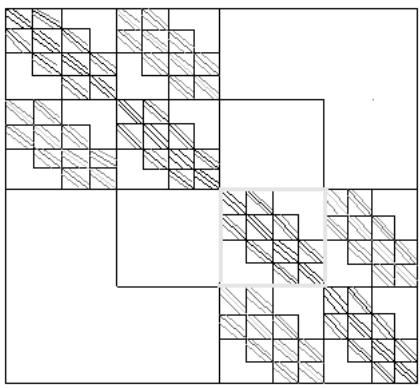

FIGURE 6. Matrix cut on 3D layer (0 order)

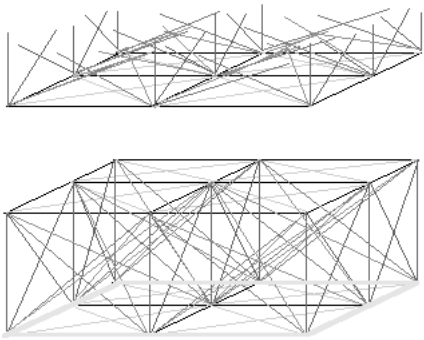

FIGURE 7. Cut on 3D layer (1 order)

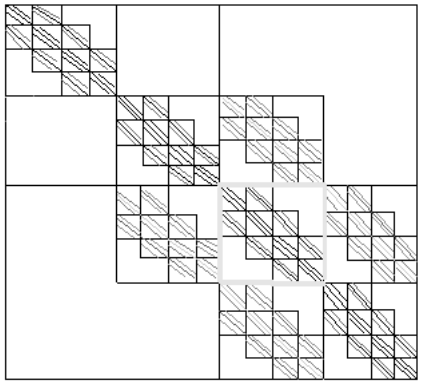

FIGURE 8. Matrix cut on 3D layer (1 order)

The first layer steps (up to a certain level) hence consist in the estimation of the Schur correction term, which in the first-order approximation is $-LM^{-1}L^H$ in which M and L again have structure, in our example they are block tridiagonal, with blocks that are tridiagonal with scalar entries. A central step is then the computation of M^{-1} with such a structure. One way of proceeding is to assume semi-separability for this layer of computation, motivated by the fact that the 2D case is (extremely) well approximated by such systems, in contrast to 3D systems. Assuming low-order semi-separability, S_k for $k = 1, \ldots, n$ will also be semi-separable, and we can calculate the Schur complements S_k using the state space theory expounded in Sections 4 and 5. In the experiments we report on here, we use a rather global semi-separable strategy, restricting the semi-separable order to $n_{kp} = rm$ (i.e., the size of a block) with a low value of r (called MRsize for 'Model Reduction Size'). It turns out that with very low values the approximation error is already negligible.

Simulation results

To characterize the various experiments we introduce the notation

'3D(level,order)2D(level,order,MRsize)'

to indicate an experiment in which the indicated levels and orders have been used in the 3D block matrix, respect. 2D block submatrices, with an eventual semi separable model order reduction size in the 2D case. Figure 9 shows $\|I - \Phi P\|_F$ for a number of situations (if a 2D specification is not shown, it means that the 2D calculation has been done on the full matrix without approximations). Each schema is carried out on orders from 0 to 3 (the latter meaning a regression of 3 layers). The X-axis shows the 3D level, while y-axis shows $\|I - \Phi P\|_F$, where the preconditioner $P \approx \Phi^{-1}$ is computed as

$$P = \begin{pmatrix} I & 0 \\ \Phi_{21}\Phi_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \Phi_{11} & \Phi_{12} \\ 0 & S^{(3D)} \end{pmatrix}$$

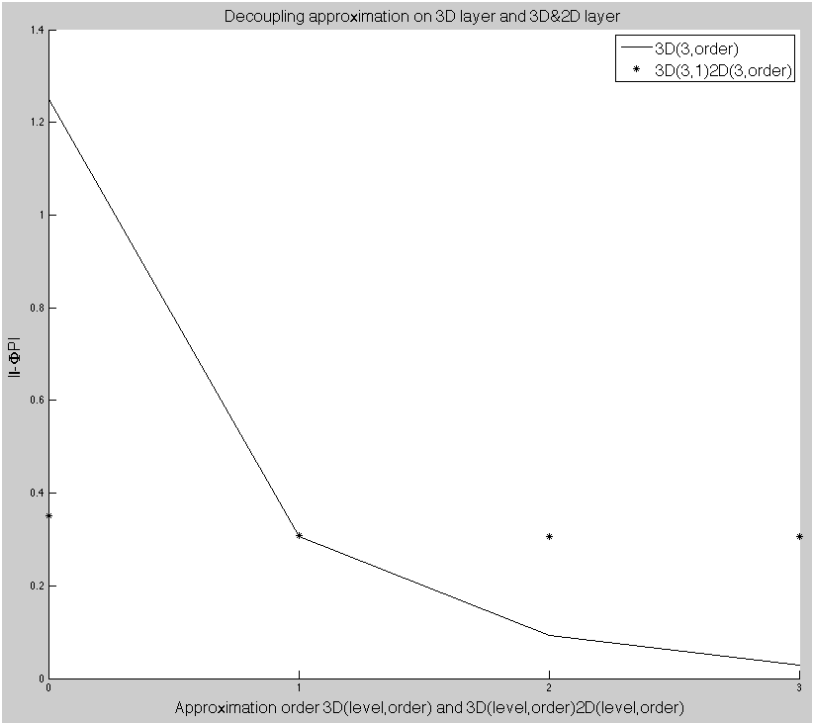

FIGURE 9. $\|I - \Phi P\|_F$ for decoupling approximation on 3D layer and 3D & 2D layers, where $3D(level, order)$ is by only decoupling on 3D layer with decoupling level *level* and approximation order *order*, while $3D(level, order)2D(level, order)$ is by decoupling on 2D layer when the *level* and *order* on 3D layer is 3 and 1

in which $S^{(3D)}$ is the approximated Schur complement by the difference approximation schemas described above. The order on the 3D layer in this figure is just one, and three levels approximation are shown as explained before. So the x-axis shows the level used on the 2D submatrices. We compare the different approximation orders on 2D taking the one without 2D approximation as a reference. As expected, cutting on both the 2D and 3D layers gives more error, but the influence of the approximation on the lower layer is much smaller compared with that on the 3D layer. This means that for a good preconditioner, the top layer must use a higher order of approximation than the lower layers. This situation is expected and it is quite pronounced.

Figure 10 shows the approximations in Frobenius norm, if in addition model order reduction with sizes 1 and 2 are applied on the 2D submatrices, and it shows the comparison with the no model reduction case. Here $r = 1$ means the realization matrices considered size 8×8, while they are 16×16 when $r = 2$.

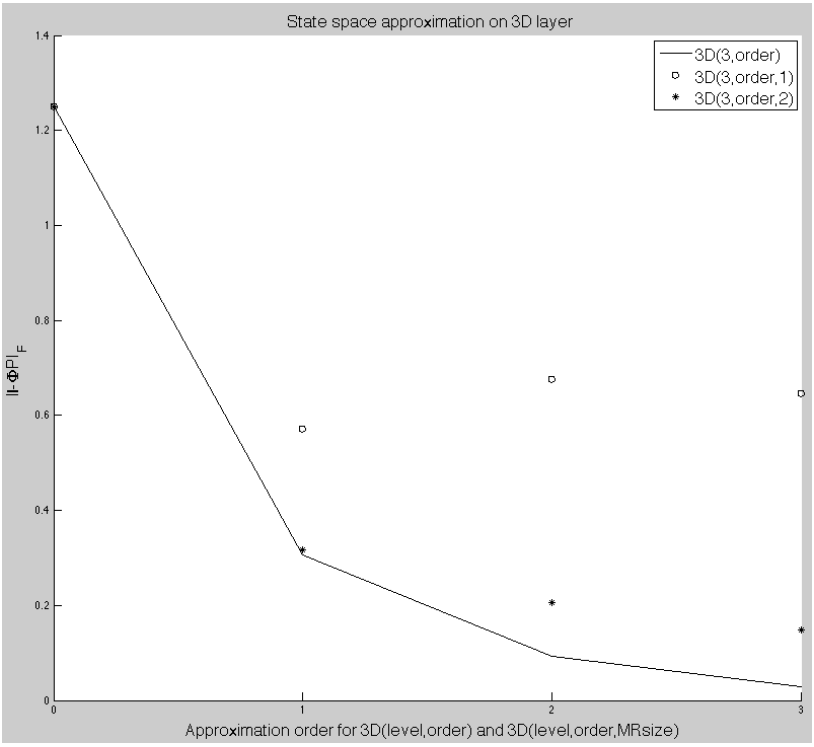

FIGURE 10. $\|I - \Phi P\|_F$ for state space approximation on 3D layer, where $3D(\text{level}, \text{order})$ is by matrix computation and

$$3D(\text{level}, \text{order}, \text{reduction rate})_{\text{state space}}$$

is by state space realization with model reduction rate *state space*

Eigenvalues of the preconditioning error. the preconditioner P is supposed to approximate Φ^{-1}. Its performance can be checked by evaluating the maximum eigenvalue $\lambda_{\max}$ of $(I-\Phi P)$ for all the proposed cases. These are listed below from table 1 to 5 for approximation schemas "Fixed point", "Cutting on 3D layer", "Cutting on 3D layer in state space with model reduction rate $r = 2$", "Cutting on 3D layer but keeping the diagonal values on off-diagonal 3D blocks", and "Cutting on both 2D and 3D layer". In these tables *level* stands for the number of level cut in 3D/2D layer; and *order* stands for the approximation order.

$\lambda_{\max}(I - \Phi P)$
0.16

TABLE 1. Fixed point

$\lambda_{\max}(I - \Phi P)$			
Order	level=1	level=2	level=3
0	0.3636	0.4420	0.5413
1	0.1304	0.1629	0.2154
2	0.0440	0.0554	0.0754
3	0.0119	0.0178	0.0244

TABLE 2. Cutting on 3D layer

$\lambda_{\max}(I - \Phi P)$			
Order	level=1	level=2	level=3
0	0.3636	0.4420	0.5413
1	0.1304	0.1629	0.2154
2	0.0440	0.0554	0.0754
3	0.0119	0.0178	0.0244

TABLE 3. Cutting on 3D layer with rate $r = 2$

$\lambda_{\max}(I - \Phi P)$			
Order	level=1	level=2	level=3
0	0.3205	0.3935	0.4884
1	0.1145	0.1428	0.1902
2	0.0381	0.0485	0.0658
3	0.0098	0.0153	0.0212

TABLE 4. Partially cutting on 3D layer

If we require the preconditioner to produce error eigenvalues smaller than 0.1, then schemas that qualify are "Cutting on 3D layer", "Cutting on 3D layer

$\lambda_{\max}(I - \Phi P)$				
Order	level=0	level=1	level=2	level=3
0	0.2154	0.2218	0.2292	0.2419
1	0.2154	0.2157	0.2160	0.2165
2	0.2154	0.2155	0.2155	0.2155
3	0.2154	0.2154	0.2154	0.2155

TABLE 5. Cutting on both 2D and 3D layer

$\lambda_{\max}(I - \Phi P)$			
Order	level=1	level=2	level=3
0	0.3636	0.4420	0.5413
1	0.1304	0.1629	0.2154
2	0.0440	0.0554	0.0754
3	0.0119	0.0178	0.0244

TABLE 6. Cutting 3D layer & 2D$\approx$ SSS

in state space with model reduction" and "Cutting on 3D layer but keeping the diagonal values on off-diagonal 3D blocks" when the approximation order is larger than one.

8. Discussion

Many more numerical results are available than those presented in this paper. We have tried in particular to use the fixed point solution as preconditioner with excellent results (almost no approximation error). If such a strategy is desired, then an efficient method must be devised to compute the fixed point solution – a question that is solvable using the exact solutions presented here, but it goes beyond the purposes of the present paper. The numerical results that we do present show that there exist very good schemas in which the preconditioning error (defined as $\lambda_{\max}(I - \Phi P)$) is below .1 or .2, resulting in very good low complexity preconditioners. This is to some extent due to the good conditioning of the Poisson matrix Φ. In future work we wish to investigate whether these properties can be extended to system matrices that are much less well conditioned, such as matrices resulting from full Maxwell equations. This will be the next effort on the program.

References

[1] S. Chandrasekaran, M. Gu, and T. Pals. Fast and stable algorithms for hierarchically semi-separable representations. In *Technical Report*. University of California at Santa Barbara, April 2004.

[2] A.-J. van der Veen. Approximant inversion of a large semi-separable matrix. In *Proc. MTNS'*04, 2004.

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

[4] P. Dewilde, K. Diepold, and W. Bamberger. A semi-separable approach to a tridiagonal hierarchy of matrices with application to image flow analysis. In *Proceedings MTNS*, 2004.

[5] P. Dewilde and A.-J. van der Veen. *Time-varying Systems and Computations*. Kluwer, 1998.

[6] Carl D. Meyer. *Matrix Analysis and Applied Linear Algebra*. SIAM, 2000.

[7] P. Dewilde, S. Chandrasekaran and M. Gu. On the numerical rank of the off-diagonal blocks of Schur complements of discretized elliptic pde's. *submitted*, March, 2007.

[8] S. Chandrasekaran, P. Dewilde, M. Gu, T. Pals, A.-J. van der Veen and J. Xia. *A fast backward stable solver for sequentially semi-separable matrices*, volume HiPC202 of *Lecture Notes in Computer Science*, pages 545–554. Springer Verlag, Berlin, 2002.

[9] J. Zhang. A class of multilevel recursive incomplete lu preconditioning techniques. *Korean J. Comp. Appl. Math.*, 8(2):213–234, 2001.

Patrick Dewilde and Haiyan Jiao
Circuits and Systems
Faculty of EEMCS
Mekelweg 4
Delft University of Technology
NL-2628 CD Delft, The Netherlands
e-mail: p.dewilde@ewi.tudelft.nl
h.jiao@tudelft.nl

Shiv Chandrasekaran
Department of Electrical and Computer Engineering
University of California,
Santa Barbara
Santa Barbara, CA 93106, USA
e-mail: shiv@ece.ucsb.edu