Weierstraß-Institut
für Angewandte Analysis und Stochastik
Leibniz-Institut im Forschungsverbund Berlin e. V.

Preprint ISSN 2198-5855

A multilevel Schur complement preconditioner for complex symmetric matrices

Rainer Schlundt

submitted: November 29, 2017

Weierstrass Institute
Mohrenstr. 39
10117 Berlin
Germany
E-Mail: rainer.schlundt@wias-berlin.de

No. 2452
Berlin 2017

2010 *Mathematics Subject Classification.* 65F08, 65F15, 65N22, 65Y05.

Key words and phrases. Complex symmetric sparse linear system, Schur complement, multilevel preconditioner, domain decomposition, low rank approximation.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Leibniz-Institut im Forschungsverbund Berlin e. V.
Mohrenstraße 39
10117 Berlin
Germany

Fax: +49 30 20372-303
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

A multilevel Schur complement preconditioner for complex symmetric matrices

Rainer Schlundt

Abstract

This paper describes a multilevel preconditioning technique for solving complex symmetric sparse linear systems. The coefficient matrix is first decoupled by domain decomposition and then an approximate inverse of the original matrix is computed level by level. This approximate inverse is based on low rank approximations of the local Schur complements. For this, a symmetric singular value decomposition of a complex symmetric matix is used. Using the example of Maxwell's equations the generality of the approach is demonstrated.

1 Introduction

We consider iterative methods for solving large sparse systems

$$Ax = b, \tag{1}$$

where $A \in \mathbb{C}^{n\times n}$, $A = A^T$, $A \neq A^H$, $b \in \mathbb{C}^n$, and $x \in \mathbb{C}^n$. Krylov subspace methods combined with a preconditioner solve the above system (1). For example, left preconditioning consists of modifying the original system into the system $M^{-1}Ax = M^{-1}b$. The preconditioner M is an approximation to A. The solve of the preconditioned system is relatively inexpensive.

The domain decomposition (DD) approach decouples the original matrix A. We do not form the global Schur complement system and do not solve it exactly. Let A be partitioned in 2×2 block form as

$$A = \begin{pmatrix} B & E \\ E^T & C \end{pmatrix} , \tag{2}$$

where $B \in \mathbb{C}^{m\times m}$, $C \in \mathbb{C}^{s\times s}$, $E \in \mathbb{C}^{m\times s}$, and $n = m + s$. We will receive the following basic block factorization of (2)

$$A = \begin{pmatrix} B & E \\ E^T & C \end{pmatrix} = \begin{pmatrix} I & 0 \\ E^T B^{-1} & I \end{pmatrix} \begin{pmatrix} B & 0 \\ 0 & S \end{pmatrix} \begin{pmatrix} I & B^{-1}E \\ 0 & I \end{pmatrix} , \tag{3}$$

where $S \in \mathbb{C}^{s\times s}$, $S = C - E^T B^{-1} E$, is the Schur complement. Using

$$A^{-1} = \begin{pmatrix} I & -B^{-1}E \\ 0 & I \end{pmatrix} \begin{pmatrix} B^{-1} & 0 \\ 0 & S^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -E^T B^{-1} & I \end{pmatrix} , \tag{4}$$

the original system (1) can be easily solved if S^{-1} is available. The goal is to show that $S^{-1} \approx C^{-1} + LRA = \tilde{S}^{-1}$, where LRA stands for low rank approximation matrix. The preconditioner M then has the following form

$$M = \begin{pmatrix} I & 0 \\ E^T B^{-1} & I \end{pmatrix} \begin{pmatrix} B & 0 \\ 0 & \tilde{S} \end{pmatrix} \begin{pmatrix} I & B^{-1}E \\ 0 & I \end{pmatrix} . \tag{5}$$

DOI 10.20347/WIAS.PREPRINT.2452 Berlin 2017

We can write

$$S = C - E^T B^{-1} E = C^{1/2}\big(I - C^{-1/2} E^T B^{-1} E C^{-1/2}\big) C^{1/2} = C^{1/2}\big(I - G\big) C^{1/2} \tag{6}$$

and

$$S^{-1} = C^{-1/2}\big(I - G\big)^{-1} C^{-1/2} = C^{-1} + C^{-1/2} G \big(I - G\big)^{-1} C^{-1/2} \,. \tag{7}$$

The symmetric matrix $G \in \mathbb{C}^{s\times s}$ has a symmetric singular value decomposition (SSVD)

$$G = C^{-1/2} E^T B^{-1} E C^{-1/2} = W \Sigma W^T \,, \tag{8}$$

where W is a unitary matrix and $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_s)$ with nonnegative σ_i (cf. [1]). Then LRA is an approximation of $C^{-1/2} G \big(I - G\big)^{-1} C^{-1/2}$. Detailed information can be found in Section 3. In Section 2, we introduce the domain decomposition framework. Numerical experiments of a model problem are presented in Section 4.
Implementation details for a real symmetric matrix A are described in [4, 5].

2 Domain decomposition framework

An interesting class of domain decomposition methods is the hierarchical interface decomposition (HID) ordering (cf. [2]). An HID ordering can be obtained from a standard graph partitioning (cf. METIS [3]). The reordered matrix has the following multilevel recursive form :

$$A_j = P_j C_{j-1} P_j^T = \begin{pmatrix} B_j & E_j \\ E_j^T & C_j \end{pmatrix} \quad \text{and} \quad C_0 \equiv A \quad \text{for} \quad j = 1, \ldots, L \,. \tag{9}$$

P_j is a permutation matrix and L the number of levels. Each block B_j in A_j has a block-diagonal structure resulting from this HID ordering. Analogous to (2), let A_j be partitioned at level j in block form as

$$A_j = \begin{pmatrix} B_j & E_j \\ E_j^T & C_j \end{pmatrix} = \begin{pmatrix} B_{j_1} & & & E_{j_1} \\ & \ddots & & \vdots \\ & & B_{j_p} & E_{j_p} \\ E_{j_1}^T & \ldots & E_{j_p}^T & C_j \end{pmatrix} \,, \tag{10}$$

where $B_j \in \mathbb{C}^{m_j \times m_j}$ is a block-diagonal matrix, $B_j = \operatorname{diag}(B_{j_1}, \ldots, B_{j_p})$, $C_j \in \mathbb{C}^{s_j \times s_j}$, $E_j \in \mathbb{C}^{m_j \times s_j}$, $E_j^T = (E_{j_1}^T, \ldots, E_{j_p}^T)$, $B_{j_i} \in \mathbb{C}^{m_{j_i} \times m_{j_i}}$, $E_{j_i} \in \mathbb{C}^{m_{j_i} \times s_j}$, $1 \le i \le p$, $n_j = m_j + s_j$, and $m_j = m_{j_1} + \cdots + m_{j_p}$. The following diagram illustrates these dependencies:

$$\begin{pmatrix} B_1 & E_1 \\ E_1^T & C_1 \end{pmatrix} \xrightarrow{P_2} \begin{pmatrix} B_2 & E_2 \\ E_2^T & C_2 \end{pmatrix} \xrightarrow{P_3} \quad \ldots \quad \xrightarrow{P_{L-1}} \begin{pmatrix} B_{L-1} & E_{L-1} \\ E_{L-1}^T & C_{L-1} \end{pmatrix} \xrightarrow{P_L} \begin{pmatrix} B_L & E_L \\ E_L^T & C_L \end{pmatrix} \,.$$

Algorithm 1 provides a detailed illustration of the multilevel HID ordering scheme.

3 Multilevel preconditioning technique

Analogous to (3), at each level j, the factorization of A_j is determined by

$$A_j = \begin{pmatrix} B_j & E_j \\ E_j^T & C_j \end{pmatrix} = \begin{pmatrix} I & 0 \\ E_j^T B_j^{-1} & I \end{pmatrix} \begin{pmatrix} B_j & 0 \\ 0 & S_j \end{pmatrix} \begin{pmatrix} I & B_j^{-1} E_j \\ 0 & I \end{pmatrix} \,, \tag{11}$$

Algorithm 1 Basic pseudocode of multilevel HID ordering

Input: Matrix A and maximum level L.

Output: Matrices B_j, C_j , and E_j for $1 \le j \le L$.

procedure MHID
  Set $C_0 = A$ and $s_0 = n$.
  for $j = 1, \dots, L$ **do**
    Compute permutation matrix P_j and $n_j = s_{j-1}$.
    Re-sort matrix C_{j-1} by $A_j = P_j C_{j-1} P_j^T = \begin{pmatrix} B_j & E_j \\ E_j^T & C_j \end{pmatrix}$.
    $n_j = m_j + s_j$. ▷ $m_j = m_{j_1} + \dots + m_{j_p}$
    Compute B_j, $B_j = \operatorname{diag}(B_{j_1}, \dots, B_{j_p})$. ▷ $B_j \in \mathbb{C}^{m_j \times m_j}$, $B_{j_i} \in \mathbb{C}^{m_{j_i} \times m_{j_i}}$
    Compute C_j. ▷ $C_j \in \mathbb{C}^{s_j \times s_j}$
    Compute E_j, $E_j^T = (E_{j_1}^T, \dots, E_{j_p}^T)$. ▷ $E_j \in \mathbb{C}^{m_j \times s_j}$, $E_{j_i} \in \mathbb{C}^{m_{j_i} \times s_j}$
  end for
end procedure

where $S_j = C_j - E_j^T B_j^{-1} E_j$ is the Schur complement at level j. Thus

$$A_j^{-1} = \begin{pmatrix} I & -B_j^{-1} E_j \\ 0 & I \end{pmatrix} \begin{pmatrix} B_j^{-1} & 0 \\ 0 & S_j^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -E_j^T B_j^{-1} & I \end{pmatrix} \tag{12}$$

is the inverse of A_j. Analogous to (7), S_j^{-1} can be approximated by C_j^{-1} plus an approximation of $C_j^{-1/2} G_j \big(I - G_j\big)^{-1} C_j^{-1/2}$. The preconditioner M_j then has the following form

$$M_j = \begin{pmatrix} I & 0 \\ E_j^T B_j^{-1} & I \end{pmatrix} \begin{pmatrix} B_j & 0 \\ 0 & \tilde{S}_j \end{pmatrix} \begin{pmatrix} I & B_j^{-1} E_j \\ 0 & I \end{pmatrix} \tag{13}$$

and

$$C_j^{-1} = P_{j+1}^T M_{j+1}^{-1} P_{j+1} \,. \tag{14}$$

At each level j, the symmetric matrix $G_j \in \mathbb{C}^{s_j \times s_j}$ has a singular value decomposition (SVD)

$$G_j = C_j^{-1/2} E_j^T B_j^{-1} E_j C_j^{-1/2} = U_j \Sigma_j V_j^H \,, \tag{15}$$

where U_j and V_j are unitary matrices and $\Sigma_j = \operatorname{diag}(\sigma_{j_1}, \dots, \sigma_{j_{s_j}})$ the singular values with real nonnegative σ_{j_i}. For the matrix G_j there exists a unitary matrix W_j such that

$$G_j = C_j^{-1/2} E_j^T B_j^{-1} E_j C_j^{-1/2} = W_j \Sigma_j W_j^T \tag{16}$$

is an SSVD. An SSVD of a symmetric matrix can be determined from its SVD. Therefore we have to modify the singular vectors corresponding to nonzero singular values (cf. [1]).

Using the Sherman-Morrison-Woodbury (SMW) formula for $(I - G_j)^{-1}$ the matrix $G_j \big(I - G_j\big)^{-1}$ results in

$$\begin{aligned} G_j\big(I - G_j\big)^{-1} = \; & W_j \Sigma_j W_j^T \Big(I + W_j \big(I - \Sigma_j W_j^T W_j\big)^{-1} \Sigma_j W_j^T \Big) \\ = \; & W_j \Sigma_j W_j^T \Big(W_j^{-T} \Sigma_j^{-1} \big(I - \Sigma_j W_j^T W_j\big) W_j^{-1} + I \Big) \cdot \\ & W_j \big(I - \Sigma_j W_j^T W_j\big)^{-1} \Sigma_j W_j^T \\ = \; & W_j \big(I - \Sigma_j W_j^T W_j\big)^{-1} \Sigma_j W_j^T \end{aligned} \tag{17}$$

and consequently

$$C_j^{-1/2} G_j \big(I - G_j\big)^{-1} C_j^{-1/2} = C_j^{-1/2} W_j \big(I - \Sigma_j W_j^T W_j\big)^{-1} \Sigma_j W_j^T C_j^{-1/2} \\ = Z_j \big(I - \Sigma_j Z_j^T C_j Z_j\big)^{-1} \Sigma_j Z_j^T \tag{18}$$

with $Z_j = C_j^{-1/2} W_j$.
The matrix $C_j^{-1} E_j^T B_j^{-1} E_j C_j^{-1}$ has the same singular values as $C_j^{-1/2} E_j^T B_j^{-1} E_j C_j^{-1/2}$ and the columns of $C_j^{-1/2} W_j$ in (18) are the singular vectors of $C_j^{-1} E_j^T B_j^{-1} E_j C_j^{-1}$ due to the following relation

$$G_j = W_j \Sigma_j W_j^T \iff C_j^{-1/2} G_j C_j^{-1/2} = Z_j \Sigma_j Z_j^T. \tag{19}$$

Thus, the computation of a low rank approximation to $S_j^{-1} - C_j^{-1}$ (cf. (7), (18)) can be obtained by the following SSVD problem

$$C_j^{-1} E_j^T B_j^{-1} E_j C_j^{-1} = Z_j \Sigma_j Z_j^T. \tag{20}$$

Let $\tilde{\Sigma}_j$ be the k largest singular values of (20) and $\tilde{Z}_j$ the corresponding singular vectors then

$$S_j^{-1} - C_j^{-1} \approx \tilde{Z}_j \big(I - \tilde{\Sigma}_j \tilde{Z}_j^T C_j \tilde{Z}_j\big)^{-1} \tilde{\Sigma}_j \tilde{Z}_j^T \tag{21}$$

is a low rank approximation. Algorithm 2 provides a construction scheme of multilevel Schur complement preconditioners based on low rank approximations.

Algorithm 2 Construction of multilevel Schur complement preconditionres

Input: Maximum level L, rank k, and matrices B_j, C_j , and E_j for $1 \le j \le L$.
Output: Matrices $\tilde{Z}_j$ and $\tilde{\Sigma}_j$ for $1 \le j \le L$.

1: **procedure** MSCP
2: Approximate C_L^{-1}. ▷ $C_L \to C_L^{-1}$
3: **for** $j = L, \ldots, 1$ **do**
4: Approximate $B_j^{-1} = \mathrm{diag}(B_{j_1}^{-1}, \ldots, B_{j_p}^{-1})$. ▷ $B_j \to B_j^{-1}$
5: Compute the k largest singular values and the corresponding singular vectors from

$$C_j^{-1} E_j^T B_j^{-1} E_j C_j^{-1} = Z_j \Sigma_j Z_j^T.$$

▷ Call Algorithm 3 to apply C_j^{-1}

6: Set $\tilde{\Sigma}_j \leftarrow \Sigma_j$ and $\tilde{Z}_j \leftarrow Z_j$.
7: **end for**
8: **end procedure**

Using Eq. (14) we get the product of C_j^{-1} with a vector v_j in the following way:

$$y_j = C_j^{-1} v_j = P_{j+1}^T M_{j+1}^{-1} P_{j+1} v_j = P_{j+1}^T M_{j+1}^{-1} u_{j+1} = P_{j+1}^T x_{j+1}\,.$$

A recursive scheme for computing the product $x_j = M_j^{-1} u_j$ is described in Algorithm 3. The solutions associated with B_j can be performed independently for each diagonal block in B_j.
Finally, the preconditioned system

$$M^{-1} A x = M^{-1} b \quad \text{with} \quad M^{-1} = C_0^{-1} = P_1^T M_1^{-1} P_1 \tag{22}$$

is to be solved.

Algorithm 3 A recursive formula for the approximation of $x_j = M_j^{-1} u_j$

Input: Matrix A_j, vector u_j at level j, and maximum level L.
Output: Vector x_j.

```
procedure Rsc(j, A_j, u_j)
    Split u_j = (u_{j,1}, u_{j,2})^T.                                  ▷ u_j = P_j u_{j-1,2}
    Compute z_1 = B_j^{-1} u_{j,1}.
    Compute z_2 = u_{j,2} - E_j^T z_1.
    if j = L then
        Compute x_{j,2} = C_j^{-1} z_2.
    else
        Compute v = P_{j+1} z_2.
        Compute w = Rsc(j + 1, A_{j+1}, v).
        Compute x_{j,2} = P_{j+1}^T w.
    end if
    Compute x_{j,2} = x_{j,2} + \tilde{Z}_j (I - \tilde{\Sigma}_j \tilde{Z}_j^T C_j \tilde{Z}_j)^{-1} \tilde{\Sigma}_j \tilde{Z}_j^T z_2.   ▷ c.f. Eq. (21)
    Compute x_{j,1} = z_1 - B_j^{-1} E_j x_{j,2}.
    return x_j = (x_{j,1}, x_{j,2})^T.                                 ▷ x_{j-1,2} = P_j^T x_j
end procedure
```

4 Numerical experiments

Using the example of Maxwell's equations we demonstrate the generality of the approach. We obtain in vector notation the following equations in integral form:

$$\begin{aligned} &\oint_P \vec{E} \cdot \mathrm{d}\vec{l} = -\frac{\partial}{\partial t} \iint_A \vec{B} \cdot \mathrm{d}\vec{A} \qquad && \oint_P \vec{H} \cdot \mathrm{d}\vec{l} = \frac{\partial}{\partial t} \iint_A \vec{D} \cdot \mathrm{d}\vec{A} + \iint_A \vec{J} \cdot \mathrm{d}\vec{A} \\ &\oiint_S \vec{B} \cdot \mathrm{d}\vec{S} = 0 && \oiint_S \vec{D} \cdot \mathrm{d}\vec{S} = \iiint_V q \,\mathrm{d}V\,. \end{aligned} \tag{23}$$

The constitutive relations belonging to them are

$$\vec{D} = \varepsilon\,\vec{E}\,, \qquad \vec{B} = \mu\,\vec{H}\,, \qquad \vec{J} = \kappa\,\vec{E}\,. \tag{24}$$

Here, A is a surface with boundary curve P, V is a volume bounded by a surface S, and q is the volume charge density. An orthogonal dual mesh is used to discretize the Maxwell's equations using the Finite Integration Technique (FIT, [9, 10, 6]). The electric and magnetic voltages and fluxes over elemtary objects are defined as state variables in the following way:

$$\begin{aligned} e_i &= \int_{L_i} \vec{E} \cdot \mathrm{d}\vec{l} & h_j &= \int_{\tilde{L}_j} \vec{H} \cdot \mathrm{d}\vec{l} & i &= 1, \dots, n_e \\ d_i &= \iint_{\tilde{A}_i} \vec{D} \cdot \vec{n} \,\mathrm{d}\vec{A} & b_j &= \iint_{A_j} \vec{B} \cdot \vec{n} \,\mathrm{d}\vec{A} & j &= 1, \dots, n_f \\ j_i &= \iint_{\tilde{A}_i} \vec{J} \cdot \vec{n} \,\mathrm{d}\vec{A} & q_k &= \iiint_{\tilde{V}_k} q \,\mathrm{d}V & k &= 1, \dots, n_p\,. \end{aligned}$$

where $\vec{n}$ is the outward-pointing normal of the faces A_j and $\tilde{A}_i$, respectively. If all field quantities vary sinusoidally with time, the coefficient matrices of the corresponding linear systems of equation are complex, symmetric, and indefinite. Using Krylov subspace methods, (22) can be solved iteratively (cf. [7, 8]).

We consider different dimensions of the coefficient matrices of the corresponding systems of linear equations, in fact $n = 16632$, $n = 40824$, and $n = 472416$. At each level j, $j = 1, 2, \ldots$, the matrix A is partitioned into p, $p \in \{0, 2, 3, 5, 10, 15\}$, non-overlapping subsets $B_{j_i}, i = 1, \ldots, p$. At each p, $p \in \{2, 3, 5, 10, 15\}$, we compute the k, $k \in \{5, 10, 15, 20\}$, largest singular values and the corresponding singular vectors to obtain a low rank approximation. For $p = 0$ the solution is computed by [7, 8]. The same applies for the computation of the solution with the coefficient matrices B_{j_i}. From Tables 1-3, we find the numbers of iterations for the different dimensions. From Tables 4-6, we find the corresponding dimensions of the matrices B_{j_i} and C_j for $j = 1, 2, \ldots$ and $i = 1, \ldots, p$ of the considered dimensions $n = 16632$, $n = 40824$, and $n = 472416$, respectively. It can be seen that the best results have been achieved for $p = 2$ and $k \in \{5, 10, 15, 20\}$. For small dimensions, also $p = 3$ is useful. For $n = 16632$ and $n = 40824$, respectively, it is useful to choose $k \in \{5, 10\}$. For greater dimensions $k \in \{15, 20\}$ is used. Experimental results indicate that this preconditioner based on Schur complement approach is robust and can achieve savings in the iteration phase.

Table 1: The number of iterations for $n = 16632$.

number of subsets (p)	k largest singular values			
	5	10	15	20
0	157			
2	128	124	126	131
3	142	147	142	143
5	146	143	143	143
10	182	187	187	180
15	211	208	217	214

Table 2: The number of iterations for $n = 40824$.

number of subsets (p)	k largest singular values			
	5	10	15	20
0	510			
2	172	168	173	167
3	182	173	176	181
5	294	283	285	275
10	347	341	330	298
15	382	366	362	368

Table 3: The number of iterations for $n = 472416$.

number of subsets (p)	k largest singular values			
	5	10	15	20
0	918			
2	344	310	262	270
3	537	532	550	530
5	596	614	616	617
10	716	758	719	763
15	827	820	831	991

Table 4: The dimensions of the matrices B_{j_i} and C_j for $n = 16632$.

p	level j	$\dim(B_{j_i})$					$\dim(C_j)$
2	1	7775	7775				1082
	2	489	490				103
	3	41	41				21
3	1	5121	4798	4915			1798
	2	528	467	543			260
	3	82	76	68			34
5	1	2913	3326	2793	2700	2770	2130
	2	358	387	377	378	278	352
	3	57	47	63	62	47	76
10	1	1663	1663	1336	1266	1094	4017
		1161	1126	1238	1145	923	
	2	310	280	310	247	258	1201
		333	300	202	284	292	
	3	88	94	91	98	97	235
		107	95	110	90	96	
15	1	1108	1109	1109	479	787	5026
		804	605	836	636	668	
		835	577	634	665	754	
	2	252	254	214	205	284	1841
		148	163	209	177	218	
		233	221	263	176	168	
	3	94	87	92	101	81	530
		99	88	83	86	88	
		108	71	88	55	90	
	4	29	24	14	17	22	194
		27	16	16	13	31	
		20	25	27	27	28	

Table 5: The dimensions of the matrices B_{j_i} and C_j for $n = 40824$.

p	level j	$\dim(B_{j_i})$					$\dim(C_j)$
2	1	19324	19351				2149
	2	967	973				209
	3	97	99				13
3	1	12326	12174	12098			4226
	2	1319	1279	1282			346
	3	103	109	103			31
5	1	7018	7251	7043	5718	7133	6661
	2	1213	1182	1163	1096	1146	861
	3	165	160	156	141	152	87
10	1	3235	3201	3079	2481	3025	8789
		3483	4082	3250	3045	3154	
	2	731	659	657	671	668	2052
		716	726	693	644	572	
	3	184	166	191	156	186	301
		190	169	173	168	168	
	4	28	24	26	24	24	24
		29	31	31	30	30	
15	1	2435	2430	2722	1894	2017	10404
		1691	1979	1893	1957	1691	
		2048	2014	2069	1642	1848	
	2	570	537	456	533	408	2828
		512	530	593	469	361	
		520	514	518	518	537	
	3	161	154	159	132	158	586
		154	141	150	148	142	
		141	139	142	166	155	
	4	34	31	29	32	35	82
		38	35	36	33	35	
		40	28	27	37	34	

Table 6: The dimensions of the matrices B_{j_i} and C_j for $n = 472416$.

p	level j	$\dim(B_{j_i})$					$\dim(C_j)$
2	1	230148	230146				12122
	2	5930	5935				257
3	1	151607	150527	150873			19409
	2	6351	6216	6239			603
	3	197	199	193			14
5	1	89225	89134	89376	88286	85534	30861
	2	5945	6049	6064	5930	5931	942
	3	188	187	187	189	183	8
10	1	41695	43598	41198	43099	40019	52167
		41313	42221	42149	41569	43388	
	2	4818	4799	4870	4716	4929	3661
		4910	4880	4709	4934	4941	
	3	361	359	360	356	361	65
		363	360	361	359	356	
15	1	27601	25284	26192	25952	26280	68943
		28966	27093	28287	28493	26501	
		26212	25965	26983	26682	26982	
	2	4280	4186	4307	4297	4299	6020
		4146	3961	4279	4262	4087	
		4239	4213	3869	4320	4178	
	3	391	376	391	390	392	209
		381	391	387	387	382	
		388	396	387	392	380	
	4	12	14	14	13	10	8
		13	13	14	13	14	
		14	15	13	13	16	

5 Conclusions

This paper presents a preconditioning method based on a Schur complement approach with low rank approximations for solving complex symmetric sparse linear systems. This method can be both recursive and non-recursive. It tries to approximate the inverse of the Schur complement by exploiting low rank approximations. For this, a hierarchical graph decomposition reorders the matrix into a multilevel block form. On the negative side, building this preconditioner can be time consuming. A solve with the matrix B_j amounts to p local and independent solves with the matrices $B_{j_i},\ i = 1, \ldots, p$. These can be carried out by a preconditioned Krylov subspace iteration. A big part of the computations to build a preconditioner based on Schur complement approach is attractive for massively parallel machines.

References

[1] Angelika Bunse-Gerstner and William Gragg. Singular value decomposition of complex symmetric matrices. *Journal of Computational and Applied Mathematics*, 21:41–54, 1988.

[2] Pascal Henon and Yousef Saad. A parallel multistage ILU factorization based on a hierrarchical graph decomposition. *SIAM J. Sci. Comput.*, 28(6):2266–2293, 2006.

[3] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.*, 20:359–392, 1998.

[4] Ruipeng Li and Yousef Saad. Low-rank correction methods for algebraic domain decomposition preconditioners. Technical Report ys-2014-5, Dept. Computer Science and Engineering, University of Minnesota, Minneapolis, MN, 2014.

[5] Ruipeng Li, Yuanzhe Xi, and Yousef Saad. Schur complement based domain decomposition preconditioners with low-rank corrections. Technical Report ys-2014-3, Dept. Computer Science and Engineering, University of Minnesota, Minneapolis, MN, 2014.

[6] Rainer Schlundt. Regular triangulation and power diagrams for Maxwell's equations. WIAS Preprint No. 2017, Weierstraß-Institut für Angewandte Analysis und Stochastik, 2014.

[7] Rainer Schlundt, Franz-Josef Schückle, and Wolfgang Heinrich. Shifted linear systems in electromagnetics. Part I: Systems with identical right-hand sides. WIAS Preprint No. 1420, Weierstraß-Institut für Angewandte Analysis und Stochastik, 2009.

[8] Rainer Schlundt, Franz-Josef Schückle, and Wolfgang Heinrich. Shifted linear systems in electromagnetics. Part II: Systems with multiple right-hand sides. WIAS Preprint No. 1646, Weierstraß-Institut für Angewandte Analysis und Stochastik, 2011.

[9] T. Weiland. A discretization method for the solution of Maxwell's equations for six-component fields. *Electronics and Communication (AEÜ)*, 31:116–120, 1977.

[10] T. Weiland. On the unique numerical solution of Maxwellian eigenvalue problems in three dimensions. *Particle Accelerators (PAC)*, 17:277–242, 1985.