Locally Optimal Block Preconditioned Conjugate Gradient Method for Hierarchical Matrices

Peter Benner\textsuperscript{1,2} and Thomas Mach\textsuperscript{1,*}

\textsuperscript{1} Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany
\textsuperscript{2} Mathematik in Industrie und Technik, Fakultät für Mathematik, TU Chemnitz, 09107 Chemnitz, Germany

We present a method of almost linear complexity to approximate some (inner) eigenvalues of symmetric self-adjoint integral or differential operators. Using $H$-arithmetic the discretisation of the operator leads to a large hierarchical ($H$-) matrix $M$. We assume that $M$ is symmetric, positive definite. Then we compute the smallest eigenvalues by the locally optimal block preconditioned conjugate gradient method (LOBPCG), which has been extensively investigated by Knyazev and Neymeyr.

Hierarchical matrices were introduced by W. Hackbusch in 1998. They are data-sparse and require only $O(n \log^2 n)$ storage. There is an approximative inverse, besides other matrix operations, within the set of $H$-matrices, which can be computed in linear-polylogarithmic complexity. We will use the approximative inverse as preconditioner in the LOBPCG method. Further we combine the LOBPCG method with the folded spectrum method to compute inner eigenvalues of $M$. That is the application of LOBPCG to the matrix $M_\mu = (M - \mu I)^2$. The matrix $M_\mu$ is symmetric, positive definite, too.

Numerical experiments illustrate the behavior of the suggested approach.

1 Introduction

The numerical solution of the eigenvalue problem for partial differential equations is of constant interest, since it is an important task in many applications, see e.g. [4]. Hierarchical matrices, introduced by Hackbusch in 1998 [6], are a useful tool to handle discretizations of partial differential operators. The discretization of integral operators with non-local kernel function, like they occur in the boundary element method (BEM) [5], leads to dense matrices. Hierarchical matrices permit the usage of the partial low-rank structure of these matrices to reduce the storage and computational complexity, see e.g. [1, 6–8] for details. So it is beneficial to be able to compute eigenvalues of hierarchical matrices. The computation of the eigenvalues of a subclass of hierarchical matrices was investigated by the authors in [2].

2 LOBPCG for Hierarchical Matrices

Preconditioned inverse iteration has become a valid alternative for computing the smallest by magnitude eigenvalues of large scale matrices. Among those LOBPCG is optimal, since the next iterate is chosen as the optimal vector in the subspace spanned by the current iterate, the preconditioned residuum and the last update. Enlarging this search space increases the costs and do not accelerate the convergence, see [12].

Algorithm 1 shows the application of [10, Algorithm 4.1] to hierarchical matrices. Using $H$-arithmetic the matrix-vector products, the computation of the preconditioner and the application of the preconditioner can be done in linear-polylogarithmic complexity. This leads to an almost linear complexity for the computation of the $d$ smallest eigenvalues.

3 Folded Spectrum Method

Sometimes one is also interested in the inner eigenvalues of $M$. Since shifting and inverting destroy the positive definiteness of $M$, we have to use the folded spectrum method to compute the inner eigenvalues by preconditioned inverse iteration.

The folded spectrum method [13] additionally squares the shifted matrix $M$, so that $M_\mu = (M - \mu I)^2$ is used.

Lemma 3.1 Let $M \in \mathbb{R}^{n \times n}$ be symmetric positive definite and $\mu$ not an eigenvalue of $M$. Then the matrix $M_\mu$ is symmetric positive definite, too. An eigenvector $v$ of $M$ is also an eigenvector of $M_\mu$, and vice versa if all eigenvalues of $M_\mu$ are simple.

This means, we can use LOBPCG to compute the smallest eigenvalue $\lambda$ and the corresponding eigenvector $v$ of $M_\mu$. An eigenvector of $M$ is an eigenvector of $M_\mu$, too. The Rayleigh quotient $v^T M v / v^T v$ provides us the sought eigenvalue.

The combination of folded spectrum method and LOBPCG works very well for hierarchical matrices, since the $H$-arithmetic enables us to square, shift and invert an $H$-matrix in almost linear complexity. In sparse arithmetic this would not be possible or far too expensive, so we really benefit here from the use of $H$-matrices.

* Corresponding author: Email thomas.mach@googlemail.com, phone +49 391 6110 469, fax +49 391 6110 500
Algorithm 1: Hierarchical Subspace LOBPCG

Input: $M \in \mathbb{R}^{n \times n}$, $X_0 \in \mathbb{R}^{n \times d}$ with $X_0^T X_0 = I$ e.g. randomly chosen and orthogonalized.
Output: $X_p \in \mathbb{R}^{n \times d}$, $\mu \in \mathbb{R}^{d \times d}$, with $\|MX_p - X_p \mu\| \leq \epsilon$

1. $B^{-1} = (M)^{-1}$; or $L = \text{adaptive-H-Cholesky-decomposition}(M) \Rightarrow B^{-1}x = L^{-T}(L^{-1}x)$;
2. for $i := 0; \|B^{-1}R\|_F > \epsilon; i++$ do
3. \hspace{1em} $R := B^{-1}(MX_i - X_i \mu); \mu := X_i^T MX_i$;
4. Compute the d eigenvectors $V$ corresponding to the smallest eigenvalues of $\mu := [X_i, R, P_i]^T M [X_i, R, P_i]$;
5. $X_{i+1} := [X_i, R, P_i]V$ and Orthogonalize $X_{i+1}; P_{i+1} := [0, R, P_i]V$;
6. end

<table>
<thead>
<tr>
<th>$n$</th>
<th>$t$ in s (LOBPCG)</th>
<th>$t_i/t_{i-1}$</th>
<th>abs. error</th>
<th>mean value(#it)</th>
<th>$t$ in s (PC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>&lt;0.01</td>
<td>1.6418E-009</td>
<td>8.6</td>
<td>&lt;0.01</td>
<td></td>
</tr>
<tr>
<td>256</td>
<td>0.01</td>
<td>2.6839E-009</td>
<td>8.8</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>1024</td>
<td>0.02</td>
<td>3.1418E-009</td>
<td>8.4</td>
<td>0.02</td>
<td></td>
</tr>
<tr>
<td>4096</td>
<td>0.07</td>
<td>5.0893E-009</td>
<td>7.4</td>
<td>0.22</td>
<td></td>
</tr>
<tr>
<td>16384</td>
<td>0.31</td>
<td>9.0040E-009</td>
<td>6.8</td>
<td>2.11</td>
<td></td>
</tr>
<tr>
<td>65536</td>
<td>1.41</td>
<td>1.1115E-008</td>
<td>6.3</td>
<td>11.67</td>
<td></td>
</tr>
<tr>
<td>262144</td>
<td>5.98</td>
<td>1.0161E-008</td>
<td>6.0</td>
<td>61.46</td>
<td></td>
</tr>
<tr>
<td>1048576</td>
<td>30.40</td>
<td>3.8918E-008</td>
<td>5.0</td>
<td>574.28</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Numerical results FEM-series, $d = 3$, $\epsilon_{new} = 10^{-4}$, $c = 0.1$, $\mathcal{H}$-matrix accuracy $10^{-7}$, PC: $\mathcal{H}$-Cholesky decomposition, no shift.

4 Numerical Results

We use the HLib [9] for our numerical computations. We test Algorithm 1 with matrices arising from the discretization of the 2D-Laplace problem over the unit square. The matrices have 8 to 1024 inner discretization points in each direction. Table 1 shows that the algorithm is of almost linear complexity. The computation of the preconditioner, here we use the $\mathcal{H}$-Cholesky decomposition, is fairly expensive. It pays off if the preconditioner can be reused for the solution of linear systems or further eigenvalues computations.

5 Conclusions

We have seen, that LOBPCG can be used to compute the smallest eigenvalues of hierarchical matrices. Further one can use the ideas of the folded spectrum method together with the efficient $\mathcal{H}$-arithmetic to compute inner eigenvalues of an $\mathcal{H}$-matrix using LOBPCG. This can be used to compute inner eigenvalues of sparse matrices as well as of data-sparse matrices like boundary element matrices. In the case of sparse matrices the sparse eigensolvers are superior.

In the case of data-sparse matrices, that are not sparse matrices, the limitation of storage limits the use of dense eigensolvers. The data-sparse $\mathcal{H}$-arithmetic together with LOBPCG enables us to solve larger eigenproblems, that do not fit in the storage otherwise. In [3] we explain the combination of folded spectrum method and preconditioned inverse iteration [11], which is related to LOBPCG. Also further numerical examples are given in there.

References