TWO-LEVEL HIERARCHICAL BASIS PRECONDITIONERS FOR COMPUTING EIGENFREQUENCIES OF CAVITY RESONATORS WITH THE FINITE ELEMENT METHOD

PETER ARBENZ AND ROMAN GEUS

Swiss Federal Institute of Technology (ETH), Institute of Scientific Computing, CH-8092 Zurich, Switzerland

E-mail: [arbenz,geus]@inf.ethz.ch

We report on experiments conducted with the implicitly restarted Lanczos algorithm for computing a few of the lowest frequencies of standing electro-magnetic waves in resonant cavities with the finite element method. The linear systems that are caused by the shift and invert spectral transformation are solved by means of two-level hierarchical basis preconditioners.

1 Introduction

Most particle accelerators use standing waves in cavities to produce the high voltage RF (radio frequency) fields required for the acceleration of the particles. The mathematical model for these high frequency electromagnetic fields is the eigenvalue problem solving the Maxwell equations in a bounded volume. Usually, the eigenfield corresponding to the fundamental mode of the cavity is used as the accelerating field. Due to higher harmonic components contained in the RF power fed into the cavity, and, through interactions between the accelerated particles and the electromagnetic field, an excitation of higher order modes can occur. The RF engineer designing such an accelerating cavity therefore needs a tool to compute the fundamental and about ten to twenty of the following eigenfrequencies together with the corresponding electomagnetic eigenfields.

Without changing the basic structure of the problem one can assume that the metallic surfaces are perfectly conducting and that the inside of the cavity Ω is all in vacuum. The electromagnetic field in the cavity is described by the Maxwell equations[1]. After separation of time and space variables and after elimination of the magnetic field intensity the differential equations

$$\mathbf{curl}\,\mathbf{curl}\,\mathbf{e}(\mathbf{x}) = \lambda\,\mathbf{e}(\mathbf{x}), \qquad \mathbf{x} \in \Omega, \qquad \lambda := \omega^2/c^2, \tag{1a}$$

$$\operatorname{div}\mathbf{e}(\mathbf{x}) = 0, \qquad \mathbf{x} \in \Omega, \tag{1b}$$

$$\mathbf{n} \times \mathbf{e} = 0, \qquad \mathbf{x} \in \Gamma := \partial\Omega. \tag{1c}$$

are obtained for the electric field intensity $\mathbf{e}$. Equations (1) have solutions only for certain values ω called *eigenfrequencies* of the electromagnetic oscillations of the resonator. These solutions are called *eigenmodes*.

The eigenvalue problem (1) can be solved analytically for a few domains, in particular for the case where Ω is a rectangular box[2]. We conducted our numerical experiments by means of this simple model which allows to compare the numerical results to an absolute reference. Moreover, for separate sector cyclotrons like the 590 MeV ring cyclotron[3] installed at the Paul Scherrer Institute (PSI) in Villigen, Switzerland, the gross shape of a cavity is actually well approximated by a rectangular box.

The variational form of the eigenvalue problem (1) is given by[2]

$$\begin{aligned} &\textit{Find } (\lambda, \mathbf{u}) \in \mathbb{R} \times W_0 \textit{ such that } \mathbf{u} \neq \mathbf{0} \textit{ and} \\ &(\mathbf{curl}\, \mathbf{u}, \mathbf{curl}\, \mathbf{v}) = \lambda(\mathbf{u}, \mathbf{v}), \qquad \forall \mathbf{v} \in W_0. \end{aligned} \tag{2}$$

Let $L^2(\Omega)$ be the Hilbert space of square-integrable functions over the 3-dimensional domain Ω with inner product $(\mathbf{u}, \mathbf{v}) = \int_\Omega \mathbf{u}(\mathbf{x}) \cdot \mathbf{v}(\mathbf{x})\, d\mathbf{x}$ and norm $\|\mathbf{u}\|_{0,\Omega} := (\mathbf{u}, \mathbf{u})^{1/2}$. In (2), $W_0 := \{\mathbf{v} \in W \mid \operatorname{div} \mathbf{v} = 0\}$ with

$$W := \{\mathbf{v} \in L^2(\Omega)^3 \mid \mathbf{curl}\, \mathbf{v} \in L^2(\Omega)^3 \ , \ \operatorname{div} \mathbf{v} \in L^2(\Omega) \ , \ \mathbf{n} \times \mathbf{v} = 0 \text{ on } \partial\Omega\}.$$

The difficulty with (2) stems from the condition $\operatorname{div} \mathbf{v} = 0$ as it is hard to find divergence-free finite elements. Therefore, ways have been looked for to get around this *divergence-free condition*. In this process care has to be taken in order not to introduce so-called *spurious modes*, i.e. eigenmodes that have no physical meaning.

In[2,4] we considered two approaches free of spurious modes, a penalty method and a mixed method. The former was implemented with Lagrange finite elements the latter with Nédélec elements[5,6]. In this note we restrict ourselves to the penalty method which proved to be superior in our experiments. The computing time needed to get a prescribed accuracy for eigenvalues and vectors was about an order of magnitude smaller with the penalty method than with the mixed method. In the penalty method approach (2) is replaced by[7,8]

$$\begin{aligned} &\textit{For fixed } s > 0, \textit{ find } (\lambda, \mathbf{u}) \in \mathbb{R} \times W \textit{ such that } \mathbf{u} \neq \mathbf{0} \textit{ and} \\ &(\mathbf{curl}\, \mathbf{u}, \mathbf{curl}\, \mathbf{v}) + s\, (\operatorname{div} \mathbf{u}, \operatorname{div} \mathbf{v}) = \lambda(\mathbf{u}, \mathbf{v}), \qquad \forall \mathbf{v} \in W, \end{aligned} \tag{3}$$

where s is a positive, usually small parameter. The eigenmodes $\mathbf{u}(\mathbf{x})$ of (3) corresponding to eigenvalues $\lambda < \mu_1 s$ are eigenmodes of (2). μ_1 is the smallest eigenvalue of the negative Laplace operator $-\Delta$ on Ω. Therefore, $\mu_1 s$ has to be above the largest eigenvalue we want to compute.

2 Finite element formulation

We triangulate the open, bounded domain $\Omega \subset \mathbb{R}^3$ with polyhedral boundary Γ by tetrahedrons Ω_e. We approximate W by the well known Lagrange (or node-based) finite elements spaces[5],

$$W_h^{(k)} := \left\{ \mathbf{v} \in W \mid \mathbf{v}|_{\Omega_e} \in (P_k(\Omega_e))^3 \right\}, \tag{4}$$

where $P_k(\Omega_e)$ is the space of polynomials in Ω_e of degree $\leq k$. In our numerical experiments[4] we found that trial spaces made up of piecewise quadratic $(k=2)$ polynomials are superior to to trial spaces made up of piecewise linear $(k=1)$ with respect to computing cost relative to accuracy. We will therefore restrict ourselves to $W_h^{(2)}$ equipped with a hierarchical basis[9,10],

$$W_h^{(2)} = W_h^{(1)} \oplus V_h^{(2)}. \tag{5}$$

The piecewise linear polynomials in $W_h^{(1)}$ are determined by their values in the four vertices of the tetrahedron. The piecewise quadratic polynomials in $V_h^{(2)}$ are determined by the function values in the midpoints of the six edges.

Using $W_h^{(2)}$ in the Rayleigh-Ritz method for discretizing (3) gives the finite dimensional problem

$$\begin{aligned} &\textit{For fixed } s > 0, \textit{ find } (\lambda_h, \mathbf{e}_h) \in \mathbb{R} \times W_h^{(2)} \textit{ such that } \mathbf{e}_h \neq \mathbf{0} \textit{ and} \\ &(\mathbf{curl}\, \mathbf{e}_h, \mathbf{curl}\, \Psi_h) + s\,(\mathrm{div}\, \mathbf{e}_h, \mathrm{div}\, \Psi_h) = \lambda(\mathbf{e}, \Psi), \qquad \forall \Psi_h \in W_h^{(2)}. \end{aligned} \tag{6}$$

Let $\{\Phi_i\}_{i=1}^n$ be a basis of $W_h^{(2)}$ and $\mathbf{e}_h = \sum_{i=1}^n \Phi_i \xi_i$. Then (6) becomes the matrix eigenvalue problem

$$A\mathbf{x} = \lambda M \mathbf{x}, \qquad \mathbf{x} = (\xi_1, \ldots, \xi_n), \tag{7}$$

where the elements of A and M are given by

$$a_{i,j} = (\mathbf{curl}\, \Phi_i, \mathbf{curl}\, \Phi_j) + s\,(\mathrm{div}\, \Phi_i, \mathrm{div}\, \Phi_j), \qquad m_{i,j} = (\Phi_i, \Phi_j).$$

For positive s, both A and M are symmetric positive definite n-by-n matrices. If the basis functions are chosen properly, then A and M are sparse.

Let us consider a rectangular box that is divided into $m_1 \times m_2 \times m_3$ boxes each of which is in turn subdivided into six tetrahedrons. Then $\dim W_h^{(2)} \approx 24 m_1 m_2 m_3$. The matrices M and A have up to 65 and 134 nonzero elements per row, respectively. Notice that $\dim W_h^{(1)} \approx 3 m_1 m_2 m_3$. We will use $W_h^{(1)}$ in our 'coarse grid' correction in the hierarchical preconditioner of section 4.

3 The implicitly restarted Lanczos algorithm (IRL)

For computing a few, say p, eigenvalues of a sparse matrix eigenvalue problem

$$A\mathbf{x} = \lambda M\mathbf{x}, \qquad A = A^T \geq 0, \quad M = M^T > 0, \tag{8}$$

closest to a number τ it is advisable to make a shift-and-invert spectral transformation with a *shift* σ close to τ and solve[11,12]

$$(A - \sigma M)^{-1} M\mathbf{x} = \mu\mathbf{x}, \qquad \mu = \frac{1}{\lambda - \sigma}. \tag{9}$$

instead of (8). The matrix $(A - \sigma M)^{-1}M$ is M-symmetric, i.e., it is symmetric with respect to the inner product $\mathbf{x}^T M\mathbf{y}$. The spectral transformation leaves the eigenvectors unchanged. The eigenvalues of (8) close to the shift become the largest absolute of (9). In addition they are relatively well-separated which improves the speed of convergence of Krylov-type subspace methods[13]. The cost of the improved convergence rate is the necessity to solve a linear system of equations involving $A - \sigma M$.

In[2,4] we compared four algorithms for computing a few of the smallest eigenvalues of (9): (i) subspace iteration[13], (ii) the block Lanczos algorithm[11], (iii) the implicitly restarted Lanczos algorithm[14,15], and (iv) the Jacobi-Davidson algorithm[16,17]. Subspace iteration was not competitive because of its low convergence rate. The block Lanczos algorithm was performing best for problems of limited size. However, the memory space available was not sufficient for solving our largest problems. The eigensolvers that permitted solving all problem sizes were the implicitly restarted Lanczos algorithm and the Jacobi-Davidson algorithm. The former was in our experiments slightly faster than Jacobi-Davidson by about 10-20%. We therefore only consider IRL in this study. We used the Fortran subroutines from the publicly available ARPACK[18].

To overcome the limitation of the Lanczos algorithm with respect to memory consumption, Sorensen proposed an elegant way to restart the iteration process[14,15]. (These ideas apply to the Arnoldi algorithm for non-symmetric eigenvalue problems as well.)

The Lanczos iteration process (here with the shift-and-invert approach)

$$\begin{aligned} \mathbf{q}_{j+1}\beta_{j+1} = \mathbf{r}_{j+1} &= (A - \sigma M)^{-1} M\mathbf{q}_j - \mathbf{q}_j\alpha_j - \mathbf{q}_{j-1}\beta_j, \\ \mathbf{q}_i^T M\mathbf{q}_j = \delta_{ij}, &\qquad \alpha_j = \mathbf{q}_j^T M(A - \sigma M)^{-1} M\mathbf{q}_j, \end{aligned} \qquad j = 1, 2, \ldots \tag{10}$$

is executed until $j = p + k$, where k is some positive integer, often $k = p$. Complete reorthogonalization to guarantee stability is feasible as by assumption $p + k$ is not big.

Let $V_j = [\mathbf{q}_1, \ldots, \mathbf{q}_j]$ and let

$$T_j = \text{tridiag}\begin{pmatrix} & \beta_1 & & \beta_2 & \cdots & \beta_{j-1} & \\ \alpha_1 & & \alpha_2 & & \cdots & & \alpha_j \\ & \beta_1 & & \beta_2 & \cdots & \beta_{j-1} & \end{pmatrix}.$$

Then, after $p+k$ iterations the Lanczos relation

$$CV_{p+k} = V_{p+k}T_{p+k} + \mathbf{r}_{p+k+1}\mathbf{e}_{p+k}^T, \qquad C = (A - \sigma M)^{-1}M, \tag{11}$$

holds. Now, k sweeps of the QR algorithm[13] with shifts $\mu_1, \ldots, \mu_k$ are applied to T_{p+k} such that $\hat{T} = \hat{Q}^T T_{p+k}\hat{Q}$, $\hat{Q} = Q_1 \cdots Q_k$, where Q_i represents the QR sweep with shift μ_i. Multiplication of (11) by $\hat{Q}$ from the right yields

$$CV_{p+k}\hat{Q} = (V_{p+k}\hat{Q})(\hat{Q}^T T_{p+k}\hat{Q}) + \mathbf{r}_{p+k+1}\mathbf{e}_{p+k}^T\hat{Q}. \tag{12}$$

As the orthogonal matrices Q_i, $i = 1, \ldots, k$, are Hessenberg matrices[19], $\hat{Q}$ has k nonzero subdiagonals. Therefore, the last $k+1$ columns of $\mathbf{r}_{p+k+1}\mathbf{e}_{p+k}^T\hat{Q}$ (and only these) are nonzero. Let $\hat{V}_p$ be the first p columns of $V_{p+k}\hat{Q}$ and $\hat{T}_p$ the $p \times p$ principal submatrix of $\hat{T} = \hat{Q}^T T_{p+k}\hat{Q}$ and let $\hat{\mathbf{r}}_{p+1}$ be the p-th column (i.e. the first non-zero column) of $\mathbf{r}_{p+k+1}\mathbf{e}_{p+k}^T\hat{Q}$. Then

$$C\hat{V}_p = \hat{V}_p\hat{T}_p + \hat{\mathbf{r}}_{p+1}\mathbf{e}_p^T \tag{13}$$

which is the Lanczos relation obtained if p steps of the iteration (10) are executed with starting vector $\hat{V}_p\mathbf{e}_1$. In ARPACK the shifts $\mu_1, \ldots, \mu_k$ are chosen as the k eigenvalues of $\hat{T}_{p+k}$ furthest away from the desired target value τ. Then, the eigenvalues of $\hat{T}_p$ are the p eigenvalues of T_{p+k} closest to the target value.

Besides the storage for the matrices A and M, the memory requirements of the IRL algorithm are essentially the space needed to store $\mathbf{q}_j$ and $M\mathbf{q}_j$, $j = 1, \ldots, p+k$, i.e., $2n(p+k)$ floating point numbers.

4 Two-level hierarchical basis preconditioners

With ARPACK it is possible to use any algorithm to solve the indefinite system of equations $(A - \sigma M)\mathbf{x} = \mathbf{y} = M\mathbf{q}_j$ in (10). We chose the iterative solver SYMMLQ of Paige and Saunders[20], a variant of the conjugate gradients method designed to handle symmetric indefinite systems of equations that we have to expect if the shift σ is inside the spectrum of A relative to M. Preconditioners for SYMMLQ have to be positive definite.

The accuracy to which the linear system in (10) is solved has to be at least as high as the desired accuracy in the eigenvalue calculation in order that the coefficients of the Lanczos three term recurrence are sufficiently accurate[18]. In our experiments, when we computed the transformed eigenvalues to an accuracy of about ε, we set the convergence tolerance for the system solver to $\varepsilon/100$.

In [4] we experimented with ILU and SSOR preconditioners. In general, they reduced the number of iteration steps. But taking the execution times into account we obtained the best results with diagonal preconditioning. In this note we study hierarchical basis preconditioners as discussed by Bank[9], see also Brenner[21]. Hierarchical basis preconditioners are most natural if finite elements of higher orders are employed. Let us assume that the basis elements $\{\Phi_i\}_{i=1}^n$ of $W_h^{(2)}$ are arranged according to the direct sum decomposition (5). Then the equation $(A - \sigma M)\mathbf{x} = \mathbf{y}$ can be written in the form

$$\begin{pmatrix} A_{11}-\sigma M_{11} & A_{12}-\sigma M_{12} \\ A_{21}-\sigma M_{21} & A_{22}-\sigma M_{22} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{pmatrix} \tag{14}$$

where the order of the $(1,1)$ block is $n_1 = \dim W_h^{(1)}$. Let

$$D := \begin{pmatrix} A_{11}-\sigma M_{11} & 0 \\ 0 & A_{22}-\sigma M_{22} \end{pmatrix}, \qquad L := \begin{pmatrix} 0 & 0 \\ A_{21}-\sigma M_{21} & 0 \end{pmatrix}. \tag{15}$$

Preconditioning $A - \sigma M$ with D amounts to applying a block Jacobi iteration to (14). The condition number of the preconditioned system is bounded by

$$\kappa(D^{-1}(A-\sigma M)) \le \frac{1+\gamma}{1-\gamma} \tag{16}$$

where $\gamma < 1$ is the constant in the strengthened Cauchy-Schwarz inequality

$$a(\mathbf{u}, \mathbf{v}) \le \gamma a(\mathbf{u}, \mathbf{u})^{1/2} a(\mathbf{v}, \mathbf{v})^{1/2}, \qquad \forall \mathbf{u} \in W_h^{(1)}, \quad \forall \mathbf{v} \in V_h^{(2)}$$

and $a(\mathbf{u}, \mathbf{v}) = (\mathbf{curl}\,\mathbf{u}, \mathbf{curl}\,\mathbf{v}) + s\,(\operatorname{div}\mathbf{u}, \operatorname{div}\mathbf{v})$. By direct evaluation we found that $\gamma \approx 0.92$ in our examples.

We however do not actually precondition $(A - \sigma M)$ by D. We do solve the system involving $D_{11} = A_{11} - \sigma M_{11}$ directly with a multifrontal method[22] as the order of D_{11} is only $\dim W_h^{(1)} \approx n/8$. But the much larger system involving $D_{22} = A_{22} - \sigma M_{22}$ is solved by m steps of a stationary iteration, either damped Jacobi or symmetric Gauss-Seidel (i.e. SSOR with $\omega = 1$). Notice that m steps of symmetric Gauss-Seidel cost only as much as $m+1$ steps of the ordinary Gauss-Seidel iteration due to the Conrad-Wallach trick[23]. We denote the thus modified preconditioner by $\hat{D}$. If ρ denotes the convergence radius of the iteration matrix corresponding to the inner iteration, the condition number in (16) is raised to[9]

$$\kappa(\hat{D}^{-1}(A - \sigma M)) \leq \frac{1+\gamma}{1-\gamma}\frac{1}{1-\rho^m} \tag{17}$$

As ρ increases with the problem size n, the number of iteration steps is not independent of n. The use of inner iterations has only a modest effect however if ρ is small or if m is chosen large.

5 Numerical experiments

We computed the 10 smallest eigenvalues of the eigenvalue problem (6) with domain $\Omega = (0,a) \times (0,b) \times (0,c)$ with $a = 520cm$, $b = 330cm$, and $c = 77cm$. This is about the size of the cavities of the 590 MeV ring cyclotron at the Paul Scherrer Institute. The box cavity has the advantage that its eigenvalues and eigenmodes are given analytically[2]. The smallest, tenth and eleventh eigenvalue of this problem are $\lambda_1 = 1.2713 \cdot 10^{-6}$, and $\lambda_{10} = 9.6167 \cdot 10^{-6}$, and $\lambda_{11} = 1.0031 \cdot 10^{-5}$. We computed the eigenvalues to a relative accuracy $\varepsilon = 10^{-8}$. In (9) we chose the spectral shift $\sigma = 10^{-6}$ such that $A - \sigma M$ is positive definite. The penalty parameter in (3) was set to $s = 1.5$. The damping factor in the Jacobi iteration was chosen to be 2/3.

The computational results have been obtained with the 32-processor HP Exemplar X-class system at ETH Zurich. We used the machine for its large memory. The following results have been carried out on one processor. Results on a parallel implementation of the code are reported in[24].

Table 1 contains timings for four problem sizes ranging from $n = \dim W_h^{(2)} = 1105$ up to $n = 24937$. The mesh sizes were $10 \times 6 \times 1$, $15 \times 9 \times 2$, $20 \times 12 \times 2$, and $25 \times 15 \times 3$, respectively. n_1 denotes the dimension of the 'coarse space' $W_h^{(1)}$. t is the overall time to solve a problem, t_C the time spent in the coarse grid correction, and t_s the time for the smoothing. This time does *not* contain the matrix multiplications of the outer iteration (SYMMLQ). In the smallest example pure diagonal preconditioning yields the fastest solution method. As the number of iteration steps increases with the problem size the performance of diagonal preconditioning deteriorates with n.

The timings of the two-level methods in Table 1 show that despite the spectral radius ρ appearing in (17), the iteration counts are hardly affected by the problem size. This indicates that ρ does not vary much. In our examples, damped Jacobi is a more effective smoother than symmetric Gauss-Seidel. This may be surprising, as the operation count of the two methods is almost the same. However, as we start the inner iteration with the zero vector, with Jacobi, the first smoothing step is just a multiplication of a diagonal

n	n_1	preconditioner	t[sec]	$\bar{n}_{it}^{out}$	t_C	t_s
1105	90	diagonal	4.4	64.3		
		two-level Jacobi, $m=1$	22.2	44.6	17.7	0.5
		two-level Jacobi, $m=2$	22.7	43.0		2.3
		two-level Jacobi, $m=3$	24.3	42.7		3.9
		two-level sym GS, $m=1$	22.5	42.8		1.7
		two-level sym GS, $m=2$	23.0	41.9		3.2
		two-level sym GS, $m=3$	25.0	41.5		4.6
5699	604	diagonal	108.2	87.6		
		two-level Jacobi, $m=1$	87.5	44.6	20.4	3.6
		two-level Jacobi, $m=2$	112.8	42.4		32.2
		two-level Jacobi, $m=3$	132.3	41.8		53.2
		two-level sym GS, $m=1$	105.8	43.1		27.2
		two-level sym GS, $m=2$	121.0	42.2		44.3
		two-level sym GS, $m=3$	141.6	41.9		65.5
10239	1105	diagonal	248.6	100.3		
		two-level Jacobi, $m=1$	142.5	44.5	28.6	8.9
		two-level Jacobi, $m=2$	222.7	42.7		85.2
		two-level Jacobi, $m=3$	300.0	42.4		167.0
		two-level sym GS, $m=1$	210.8	42.9		61.6
		two-level sym GS, $m=2$	250.0	42.3		111.2
		two-level sym GS, $m=3$	289.1	42.1		167.3
24937	2840	diagonal	976.2	123.6		
		two-level Jacobi, $m=1$	481.6	44.3	63.3	27.0
		two-level Jacobi, $m=2$	698.1	42.0		264.4
		two-level Jacobi, $m=3$	865.1	41.4		479.3
		two-level sym GS, $m=1$	648.4	42.4		212.1
		two-level sym GS, $m=2$	825.1	41.6		386.4
		two-level sym GS, $m=3$	981.6	41.5		595.8

Table 1. Execution times and average iteration numbers for various preconditioners

matrix with a vector. In all problem sizes we observe that increasing the number m of smoothing steps lowers $\bar{n}_{it}^{out}$ however not to the extent that the additional computing time is outweighed. This indicates that multiplying a sparse matrix with a vector is a very expensive operation which is probably due to low performance caused by indirect addressing. This last observation also explains the relatively small portion of two-level preconditioning with $m=1$ at the overall computing time. Most of the overall time is spent in the computation of the residual in SYMMLQ.

Our results show that hierarchical preconditioners are well suited for the systems of equations that have to be solved in IRL. The number of iteration steps is independent of the problem size. The good performance of Jacobi indicates that the smoother does not need to access much of $A - \sigma M$ to be effective. An overlapping Schwarz smoother with small subdomains may therefore help to improve the convergence rate. The time for the coarse grid correction could be reduced by moving from the two-level to a multilevel algorithm. This however is not of interest as long as matrix times vector multiplications dominate the computing time.

References

1. J. D. Jackson. *Classical Electrodynamics*. Wiley, New York, 2nd edition, 1975.
2. St. Adam, P. Arbenz, and R. Geus. Eigenvalue solvers for electromagnetic fields in cavities. Technical Report 275, ETH Zürich, Computer Science Department, October 1997. (Available at URL `http://www.inf.ethz.ch/publications/tr.html`).
3. Information on the accelerator installation at PSI is available at URL `http://www1.psi.ch/www_psi_hn/psi-info/1-01/`.
4. P. Arbenz and R. Geus. A comparison of solvers for large eigenvalue problems originating from Maxwell's equations. *Numer. Lin. Alg. Appl.*, 5, 1998.
5. Ph. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. North-Holland, Amsterdam, 1978. (Studies in Mathematics and its Applications, 4).
6. V. Girault and P.-A. Raviart. *Finite Element Methods for the Navier-Stokes Equations*. Springer-Verlag, Berlin, 1986. (Springer Series in Computational Mathematics, 5).
7. F. Kikuchi. Mixed and penalty formulations for finite element analysis of an eigenvalue problem in electromagnetism. *Comput. Methods Appl. Mech. Eng.*, 64:509–521, 1987.
8. R. Leis. Zur Theorie elektromagnetischer Schwingungen in anisotropen Medien. *Math. Z.*, 106:213–224, 1968.
9. R. E. Bank. Hierarchical bases and the finite element method. *Acta Numerica*, 5:1–43, 1996.
10. H. Yserentant. Hierarchical bases. In R. E. O'Malley, ed., *ICIAM 91: Proceedings of the Second International Conference on Industrial and Applied Mathematics*, pages 256–276. SIAM, Philadelphia, PA, 1992.

11. R. Grimes, J. G. Lewis, and H. Simon. A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems. *SIAM J. Matrix Anal. Appl.*, 15:228–272, 1994.
12. K. Meerbergen and D. Roose. Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems. *IMA J. Numer. Anal.*, 16:297–346, 1996.
13. B. N. Parlett. *The Symmetric Eigenvalue Problem.* Prentice Hall, Englewood Cliffs, NJ, 1980.
14. D. Sorensen. Implicite application of polynomial filters in a k-step Arnoldi method. *SIAM J. Matrix Anal. Appl.*, 13:357–385, 1992.
15. D. Calvetti, L. Reichel, and D. C. Sorensen. An implicitely restarted Lanczos method for large symmetric eigenvalue problems. *Electronic Trans. Numer. Anal.*, 2:1–21, 1994.
16. G. L. G. Sleijpen and H. A. van der Vorst. A Jacobi-Davidson iteration method for linear eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 17:401–425, 1995.
17. D. R. Fokkema, G. L. G. Sleijpen, and H. A. van der Vorst. Jacobi-Davidson style QR and QZ algorithms for the partial reduction of matrix pencils. Preprint 941, revised version, Utrecht University, Department of Mathematics, Utrecht, The Netherlands, January 1997. (To appear in SIAM J. Sci. Comput.).
18. R. B. Lehoucq, D. C. Sorensen, and C. Yang. *ARPACK Users' Guide: Solution of Large Scale Eigenvalue Problems by Implicitely Restarted Arnoldi Methods.* Department of Mathematical Sciences, Rice University, Houston TX, October 1997. (The software and this guide are available at URL `ftp://ftp.caam.rice.edu/pub/software/ARPACK/`).
19. G. H. Golub and C. F. van Loan. *Matrix Computations.* The Johns Hopkins University Press, Baltimore, MD, 2nd edition, 1989.
20. C. C. Paige and M. A. Saunders. Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.*, 12:617–629, 1975.
21. S. C. Brenner. Preconditioning complicated finite elements by simple finite elements. *SIAM J. Sci. Comput.*, 17:1269–1274, 1996.
22. C. Ashcraft and R. Grimes. The influence of relaxed supernode partitions on the multifrontal method. *ACM Trans. Math. Softw.*, 15:291–309, 1989.
23. J. Ortega. *Introduction to Parallel and Vector Solution of Linear Systems.* Plenum Press, New York, 1998.
24. P. Arbenz and R. Geus. Parallel solvers for large eigenvalue problems originating from Maxwell's equations. In *Euro-Par '98*. Springer-Verlag, 1998. (Lecture Notes in Computer Science).