A parallelizable and fast algorithm for very large generalized eigenproblems

Henk A. van der Vorst and Gerard L.G. Sleijpen

Mathematical Institute, University of Utrecht,
Budapestlaan 6, Utrecht, the Netherlands

Abstract. We discuss a novel iterative approach for the computation of a number of eigenvalues and eigenvectors of the generalized eigenproblem $Ax = \lambda Bx$. Our method is based on a combination of the Jacobi-Davidson method and the QZ-method. For that reason we refer to the method as JDQZ. The effectiveness of the method is illustrated by a numerical example.

1 Introduction

Iterative techniques for computing eigenvalues and eigenvectors of a large generalized eigenproblem $Ax = \lambda Bx$, usually require that the system is reduced to a standard eigenproblem, for instance $B^{-1}Ax = \lambda x$ (provided that B is nonsingular). This problem can be handled by standard iteration techniques, such as Arnoldi's method [1]. More advanced techniques apply clever implementations of the Arnoldi method with a shift and invert approach:

$$(A - \sigma B)^{-1}Bx = \tilde{\lambda}x,$$

in which σ is chosen close to the desired eigenvalue λ. For an elegant computational approach along these lines, see [7]. Another example of such an approach for problems that we are interested in, see Section 2, is given in [3].
The disadvantage of these conventional approaches is that in each iteration step of the eigensolver a matrix has to be inverted, or, more precisely, that a linear system has to be solved to high accuracy. For very large problems, with sparse matrices A and B, this may pose severe limitations on the order of the matrices for which computation is still feasable.

In our presentation we will describe a new iterative technique for the computation of a number of selected solutions for the eigenproblem

$$(A - \lambda B)x = 0,$$

with A, B general large sparse $(n \times n)$ matrices, in which it is avoided to transform the problem by an inversion operation. The technique is based on the Jacobi-Davidson method with reduction to Schur form. With this technique we are able to compute a number of eigenvalues and eigenvectors, with subspaces of restricted dimension, and without inversion of any of the matrices involved. The

standard methods of Arnoldi and Lanczos are not well-suited for this.
In the Jacobi-Davidson method a subspace is generated onto which the given eigenproblem is projected. The much smaller projected eigenproblem is solved by standard direct methods, and this leads to approximations for the eigensolutions of the given large problem. This is the 'Davidson' part of the algorithm. Then, a correction equation for a selected eigenpair is set up. The solution of the correction equation defines an orthogonal correction for the current eigenvector approximation; this is the 'Jacobi' part of the algorithm. The correction is used for the expansion of the subspace and the procedure is repeated.
In the new algorithm the large system is projected on a suitable subspace, in a way that may be viewed as a truncated and inexact form of the QR or QZ factorization. The projected system itself is reduced to Schur form by the QR or QZ algorithm. For this reason we refer to the algorithm as JDQR (=Jacobi-Davidson-QR), when $B = I$, and as JDQZ in the general case.

As we will see, the method lends itself perfectly well for parallel execution.

2 An example of very large eigenproblems

Although large eigenproblems arise in many scientific problems, we have been particularly motivated recently by MHD-problems. In particular, we study the dynamics of plasmas in a magnetic field with computational models. The results are applied for further understanding of the stability of Tokomak fusion reactors and of coronal loops, as well as of solar flares.
The interaction of plasma and a magnetic field is governed by essentially the flow equations for fluids combined with the Maxwell equations, and this system has the form

$$\boxed{\begin{aligned}
\frac{\partial \rho}{\partial t} &= -\nabla \cdot (\rho \mathbf{V}) \\
\rho \frac{\partial \mathbf{V}}{\partial t} &= -\rho(\mathbf{V} \cdot \nabla)\mathbf{V} - \nabla p \\
&\quad +(\nabla \times \mathbf{B}) \times \mathbf{B} \\
\frac{\partial p}{\partial t} &= -(\mathbf{V} \cdot \nabla) p - \gamma p \nabla \cdot \mathbf{V} + \\
&\quad (\gamma - 1)\eta(\nabla \times \mathbf{B})^2 \\
\frac{\partial \mathbf{B}}{\partial t} &= \nabla \times (\mathbf{V} \times \mathbf{B}) - \nabla \times (\eta \nabla \times \mathbf{B}) \\
\\
\text{with} \quad & \nabla \cdot \mathbf{B} = 0.
\end{aligned}}$$

The last equation is considered as an initial condition on $\mathbf{B}$. One of the approaches, taken in our project, is to consider small perturbations of the unknowns with respect to some known equilibrium, and this leads to a linear system for

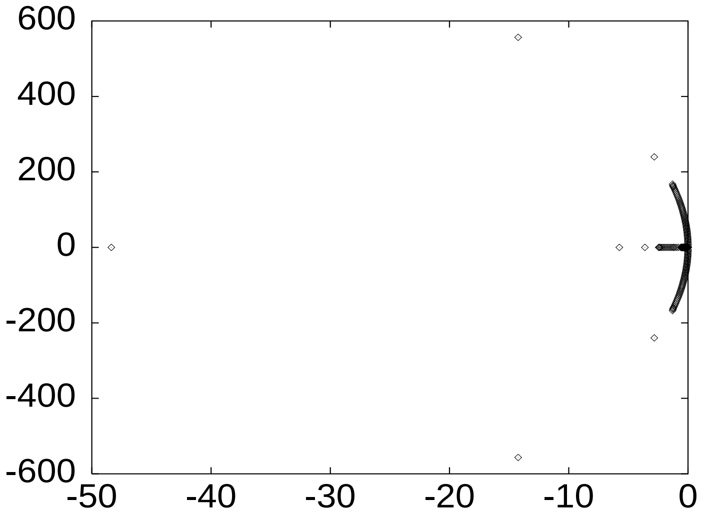

Fig. 1. Eigenvalues for MHD problem

the first order perturbations. The solution for this linearized system is assumed to be of the form $e^{\lambda t}$, and this leads to a large linear generalized eigenproblem. Due to the kind of discretization we use, partly finite differences, and partly Fourier series, this leads to block tridiagonal matrices, with typically $100 - 2,000$ blocks, of size $80 - 320$ each. This amounts to matrices of orders in the range $8,000 - 640,000$.

The matrices have eigenvalues that can be grouped in very large ones (associated with 'Sound waves'), very small ones, and intermediate ones (associated with 'Alfvén' waves), and we are interested in the Alfvén spectrum. In Figure 1 we see the entire spectrum, the middle part of Figure 2 shows the 'Alfvén' spectrum (note the different scales).

The order and the structure of these matrices makes standard direct methods unpractical, and therefore we consider iterative methods. Similar problems have been solved up to orders of a few tens of thousands with the methods of Lanczos (with a code of Cullum et al [3]) and by Shift and Invert Implicitly Restarted Arnoldi method [7]. The applicability of these methods is limited, due to the fact that inversion of matrices is required, and this becomes too expensive for the very large problems that we are interested in. As we will see, inversion of matrices is essential for these methods in order to obtain convergence for the Alfvén spectrum.

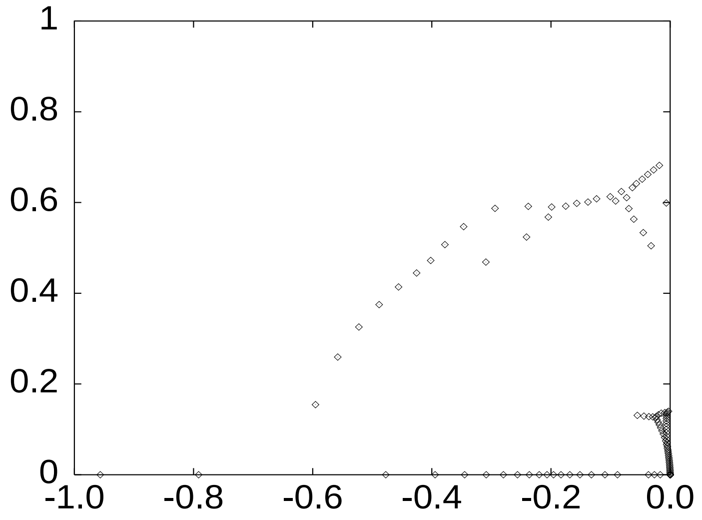

Fig. 2. Alfvén part of spectrum

3 The Jacobi-Davidson method

In this section, we consider the generalized eigenproblem $Ax = \lambda Bx$, and we want to compute a few eigenvalues close to a given target $\tau \in \mathbb{C}$. Suppose that we have a suitable low-dimensional subspace, spanned by $v_1, \ldots, v_m$, and we want to identify good eigenvector approximations in this subspace. Let V_m denote the $n \times m$ matrix with v_i as its i-th column. Let us further assume that the vectors v_i form an orthonormal basis. Then the Petrov-Galerkin approach is to find a vector $y \in \mathbb{R}^m$, and a θ, such that

$$AV_m y - \theta BV_m y \perp \{w_1, \ldots, w_m\},$$

for some convenient set of vectors $\{w_j\}$. We will assume that these vectors are also orthonormal and that they form the columns of a matrix W_m. This leads to an eigenproblem for projected matrices of order m:

$$W_m^* AV_m y - \theta W_m^* BV_m y = 0, \tag{1}$$

where W_m^* denotes the adjoint of W_m. In applications we have that $m \ll n$. The solutions θ are referred to as the (Petrov-)Ritz values (approximations for eigenvalues of $A - \lambda B$), and $V_m y$ is the corresponding (Petrov-)Ritz vector, with respect to the subspace spanned by $v_1, \ldots, v_m$.

The main problem is to find the expansion vectors v_{m+1} and w_{m+1} for the next iteration step in the process for improving the eigen approximations. In [6], it is shown how the expansion vectors can be selected for the standard eigenproblem

$Ax = \lambda x$, and in [4] the same approach is followed for generalized eigenproblems. For this approach it is not necessary to transform the given generalized eigenproblem to a standard one (for instance, by inverting one of the matrices A or B, or a combination).

We will briefly describe this inverse-free Jacobi-Davidson approach, for details we refer to [4]. First we use the QZ algorithm [5] to reduce (1) to a generalized Schur form. With m the dimension of span$\{v_j\}$, this QZ computation delivers orthogonal $(m \times m)$ matrices U_R and U_L, and upper triangular $(m \times m)$ matrices S_A and S_B, such that

$$U_L^* \, (W_m^* A V_m) \, U_R = S_A \,, \tag{2}$$

$$U_L^* \, (W_m^* B V_m) \, U_R = S_B \,. \tag{3}$$

The decomposition is ordered so that the leading diagonal elements of S_A and S_B represent the eigenvalue approximation closest to the target value τ. The approximation for the eigenpair is then taken as

$$(\tilde{q}, \theta) \equiv (V_m U_R(:,1), S_B(1,1)/S_A(1,1)\,), \tag{4}$$

assuming that $S_A(1,1) \neq 0$ (see [4] for further details). With this approximation for the eigenvector and the eigenvalue, we compute the *residual vector*:

$$r \equiv A\tilde{q} - \theta B\tilde{q}.$$

We also compute an auxiliary vector $\gamma\tilde{z} = A\tilde{q} - \tau B\tilde{q}$, where γ is such that $||\tilde{z}||_2 = 1$.
With these two vectors we can define the correction equation for $\tilde{q}$, that is so typical for the Jacobi-Davidson approach: we want to find the correction v for $\tilde{q}$ with

$$\tilde{q}^* v = 0 \;\text{ and }\; (I - \tilde{z}\tilde{z}^*)(A - \theta B)(I - \tilde{q}\tilde{q}^*)v = -r\,. \tag{5}$$

In practice, we solve (5) only approximately by a few steps of, for instance, GMRES (sometimes with a suitable preconditioner). Then we orthonormalize the approximation for v with respect to $v_1, \ldots, v_m$, and take the results as our new v_{m+1}. For the expansion vector w_{m+1}, we take the vector $Av - \tau Bv$, orthonormalized with respect to $w_1, \ldots, w_m$.
It can be shown that in this case $\tilde{z} = W_m U_L(:,1)$.

The above sketched algorithm descibes the *Harmonic Petrov value approach* proposed in [4].

4 The Jacobi-Davidson QZ-method

In some circumstances the Jacobi-Davidson method has apparent disadvantages with respect to other iterative schemes, for instance the shift-and invert Arnoldi method [7]. For instance, in many cases we see rapid convergence to one single eigenvalue, and what to do if we want more eigenvalues? For Arnoldi this is not a big problem, since the usually slower convergence towards a particular eigenvalue goes hand in hand with simultaneous convergence towards other eigenvalues. So after a number of steps Arnoldi produces approximations for several eigenvalues. For Jacobi-Davidson the obvious approach would be to restart with a different target, with no guarantee, hoewever, that this leads to a new eigenpair. Also the detection of multiple eigenvalues is a problem, but this problem is shared with the other subspace methods.

A well-known way out of this problem is to use a technique, known as *deflation*. If an eigenvector has converged, then we continue in a subspace spanned by the remaining eigenvectors. A problem is then how to re-use information obtained in a previous Jacobi-Davidson cycle. In [4] an algorithm is proposed by which several eigenpairs can be computed. The algorithm is based on the computation of a partial generalized Schur form for the matrix pair (A, B):

$$AQ_k = Z_k S_k \quad \text{and} \quad BQ_k = Z_k T_k,$$

in which Q_k and Z_k are n by k orthonormal matrices and S_k, T_k are k by k upper triangular matrices, with $k \ll n$. Note that if (x, λ) is an eigenpair of $S_k x - \lambda T_k x = 0$, then $(Q_k x, \lambda)$ is an eigenpair of $A - \lambda B$.

We now proceed in the following way in order to obtain this partial generalized Schur form for eigenvalues close to a target value τ.

Step I: Given orthonormal subspace bases $v_1, \ldots, v_i$, with matrix V_i, and w_1, ..., w_i, with matrix W_i, we compute the projected matrices $M_A = W_i^* A V_i$ and $M_B = W_i^* B V_i$. For these $i \times i$ matrices M_A and M_B, we compute the complete generalized Schur form

$$M_A U_R = U_L S_A \quad \text{and} \quad M_B U_R = U_L S_B, \tag{6}$$

in which U_R and U_L are orthonormal, and S_A and S_B are upper triangular. This can be done with the standard QZ algorithm [5].
Then we orden S_A, and S_B such that the $|S_B(i,i)/S_A(i,i) - \tau|$ form a nondecreasing row for increasing i. The first few diagonal elements of S_A and S_B then represent the eigenapproximations closest to τ, and the first few of the correspondingly reordered columns of V_i represent the subspace of best eigenvector approximations. If memory is limited then this subset can be used for restart, that is the other columns are simply discarded. The remaining subspace is expanded according to the Jacobi-Davidson method.
After convergence of this procedure we have arrived at an eigenpair (q, λ) of $A - \lambda B$: $Aq = \lambda Bq$. The question is how to expand this partial generalized Schur form of dimension 1. This will be shown in step II.

Step II: Suppose we have already a partial Schur form of dimension k, and we want to expand this by an appropriate new column q:

$$A\,[Q_k, q] = [Z_k, z] \begin{bmatrix} S_k & s \\ & \alpha \end{bmatrix}$$

and

$$B\,[Q_k, q] = [Z_k, z] \begin{bmatrix} T_k & t \\ & \beta \end{bmatrix}.$$

After some standard linear algebra manipulations it follows that

$$(I - Z_k Z_k^*)(\beta A - \alpha B)(I - Q_k Q_k^*)q = 0 \;\text{ and }\; Q_k^* q = 0,$$

which expresses that the new pair $(q, \lambda \equiv \alpha/\beta)$ is an eigenpair of the matrix pair

$$(I - Z_k Z_k^*)A(I - Q_k Q_k^*),\; (I - Z_k Z_k^*)B(I - Q_k Q_k^*). \tag{7}$$

Approximations for an eigenpair of this matrix pair can be computed by applying the Jacobi-Davidson algorithm (with Schur form reduction, as in step I), with appropriate W and V. Expansion vectors for V and W are computed by solving approximately the Jacobi-Davidson correction equation:

$$(I - \tilde{z}\tilde{z}^*)(I - Z_k Z_k^*)(A - \lambda B)(I - Q_k Q_k^*)(I - \tilde{q}\tilde{q}^*)v = -r, \tag{8}$$

with $Q_k v = 0$, and $\tilde{q}^* v = 0$. This leads to projected matrices M_A and M_B, for which it is easy to prove that:

$$M_A = W^*(I - Z_k Z_k^*)A(I - Q_k Q_k^*)V = W^* A V, \tag{9}$$

and

$$M_B = W^*(I - Z_k Z_k^*)B(I - Q_k Q_k^*)V = W^* B V. \tag{10}$$

For these matrices we construct again, as in step I, the generalized Schur decomposition:

$$M_A U_R = U_L S_A \;\text{ and }\; M_B U_R = U_L S_B.$$

Upon convergence of the Jacobi-Davidson iteration, we obtain the expansion vector $q = V U_R(:, 1)$ for Q_k, and the expansion vector z for Z_k is computed with the relation $z = W U_L(:, 1)$.

Some notes are appropriate:

1. In our computations, we have used the explicitly deflated matrix $A - \lambda B$ in (8), whereas we could have exploited the much less expensive expressions (9) and (10). We have made this choice for stability reasons. Despite these more expensive computations, it is shown in [4], by numerical experiments, that the entire procedure leads to a very efficient computational process. An explanation for this is that after convergence of some eigenvectors, the pair M_A, M_B will be better conditioned, so that the correction equation (8) in the Jacobi-Davidson step is more easily solved.
2. The correction equation (8) may be solved (approximately) by a preconditioned iterative solver, and it is shown in [4] that the same preconditioner can be used with great efficiency for different eigenpairs. Hence, the costs for constructing accurate preconditioners can be spread over many systems to be solved.

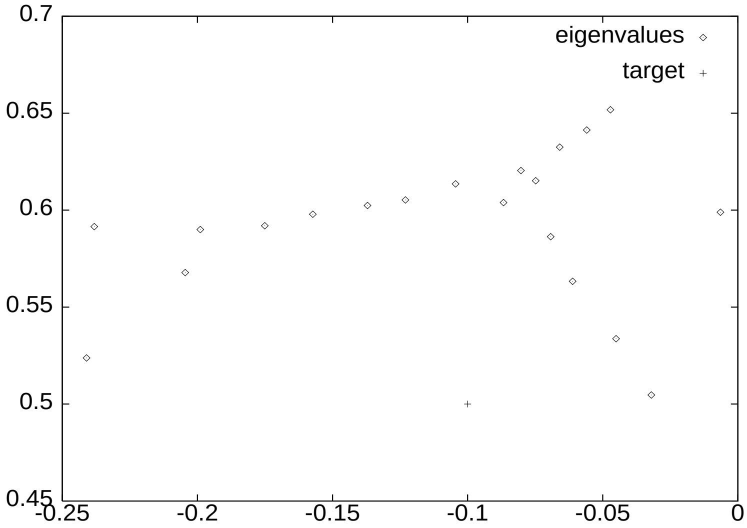

Fig. 3. Computed eigenvalues

5 An example

We will now briefly describe results for the sketched approach for a generalized eigenproblem $Ax = \lambda Bx$, associated to the MHD problems discussed in Section 2. In view of the problems that we actually want to solve, our example is just a 'toy problem' of very modest dimension, $n = 416$. For more information on this particular problem, see for instance [2].

In Figure 1 we see the complete spectrum for this case; Figure 2 shows the so-called Alfvén part of the spectrum. Note the different scales from which it is obvious that the Alfvén spectrum is an interior part of the spectrum, and without Shift-and-Invert it is almost impossible to compute this part with Krylov subspace methods. The 20 eigenvalue approximations, that we have computed with the QZ-variant of our algorithm, are shown in Figure 3. The computations have been carried out in about 15 decimal digits accuracy on a SUN workstation. The target value, indicated in picture 3 by a '+', was $\tau = -.1 + .5i$. The maximum dimension of the subspaces for the Jacobi-Davidson part of the algorithm was fixed to 15. As soon as we arrived at that dimension, the subspace was shrinked, as described in **Step I** above, to 10. An eigenpair was considered to be converged as soon as the norm of the residual for the normalized eigenvector approximation was below 10^{-9}. For the preconditioner in the solution process for the correction equation, we used the exact inverse of $A - \tau B$ (for fixed τ). The correction equations were solved approximately, namely with only 1 step of GMRES. The complete converge history, norms of the residuals of eigenapproximations versus the total amount of floating point operations, is shown in Figure 4. Each time when the curve hits the 10^{-9} level, indicated by the dotted

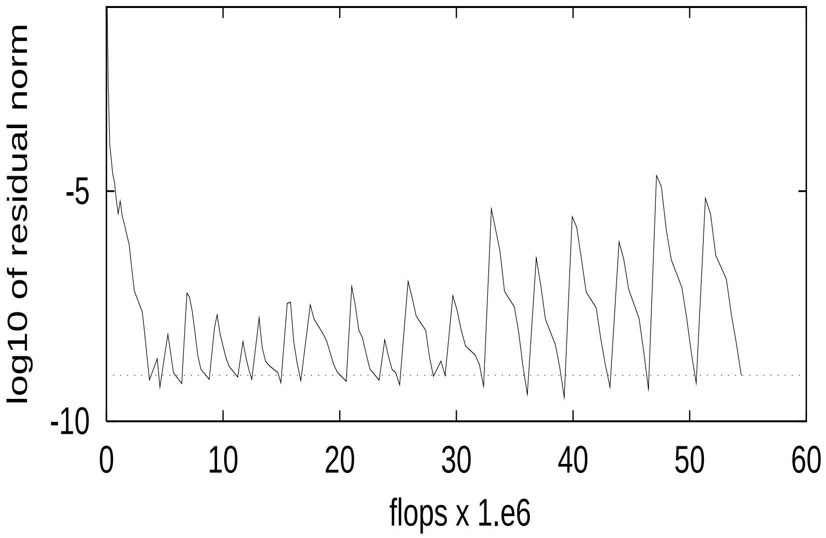

Fig. 4. Convergence history

horizontal line, the algorithm has discovered a new eigenvalue. In our case that has happened 20 times, after which we have stopped the algorithm.

6 Parallel aspects

The Jacobi-Davidson method lends itself in an excellent way for parallel computing because of its explicit nature. We will discuss the computational elements of the algorithm in view of parallel processing.

- In the initialization step subspaces V and W are chosen and orthonormal basis for these subspaces are constructed. This involves inner products and vector updates.
- In the first step of the iteration loop, the matrices M_A and M_B have to be computed. The projections with W and V, as in equations (9) and (10), also involve only inner products and vector updates. Note that it is sufficient to compute only the last recent row and colums of these matrices. This again involves a number of innerproducts: one per new matrix element.
- In the second step the eigenpairs of the projected pair are computed. The projected system is of low dimension: the projected matrices are typically of order less than 30, and hence the number of computations in this step is limited and is often negligible with respect to the number of computations in the other steps. Therefore, we let all processors do this computation, in order to avoid further communication.
- In the third step the selected Ritz vector is computed with (4) by taking a

linear combination of the columns of V. Hence, if cleverly implemented, no multiplications with the matrices A and B are needed in this step, but only vector updates.
- In the fourth step the convergence criterion is checked. This typically involves the computation of the norm of the residual r.
- The computation of the approximate solution of the correction equation (8) is usually the most expensive step in the algorithm. If this is done by a method like GMRES, then the only time-consuming operations are matrix-vector multiplications, vector updates, and inner products. Preconditioning for GMRES requires special attention with respect to parallelizability; we will not discuss this aspect in this paper.
Note that we explicitly deflate with Z_k and Q_k in (8). Multiplication with the entire operator can be carried out in five steps. First multiply with the two projectors on the right, then multiply with the matrices A and B, and take a linear combination of the results, and finally multiply with the projectors on the left. A multiplication with a projector requires only inner products and vector updates.
- In the final step of the algorithm, the subspace V is expanded with the solution of the correction equation and this new basis vector is orthogonalized against the others. This process also requires only inner products and vector updates.

From this global analysis we learn that the main CPU time-consuming operations are multiplications with matrices, vector updates, and inner products, and we only need to parallelize these operations. The inner products may require global communication (in distributed processing mode), and this may need further attention for large numbers of processors.

We have implemented this algorithm on the Cray T3D; a distributed memory computer. The communication steps have been implemented with the fast SHMEM_GET and SHMEM_PUT routines. The observed speed-ups for a slightly different formulation of the algorithm, were very satisfactory: in [8] a problem of order 274625 is described for which a linear speed-up was observed when increasing the number of processors from 16 to 64 (without preconditioning for (8)).

References

1. W. E. Arnoldi. The principle of minimized iteration in the solution of the matrix eigenproblem. *Quart. Appl. Math.*, 9:17–29, 1951.
2. J.G.L. Booten, D.R. Fokkema, G.L.G. Sleijpen, and H.A. van der Vorst. Jacobi-Davidson methods for generalized MHD-eigenvalue problems. *Zeitschrift für Angewandte Mathematik und Mechanik (ZAMM)*, 1996.
3. J. Cullum, W. Kerner, and R. Willoughby. A generalized nonsymmetric Lanczos procedure. *Computer Physics Communications*, 53:19–48, 1989.
4. 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. Technical Report Preprint 941, Mathematical Institute, Utrecht University, 1996.

5. G. H. Golub and C. F. van Loan. *Matrix Computations.* The Johns Hopkins University Press, Baltimore, 1989.
6. 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(2):401–425, 1996.
7. D. C. Sorenson. Implicit application of polynomial filters in a k-step Arnoldi method. *SIAM J. Matr. Anal. Appl.*, 13(1):357–385, 1992.
8. M.B. Van Gijzen. A parallel eigensolution of an acoustic problem with damping. submitted for publication.