CHAPTER IX

Jacobi-Davidson Methods

38 Davidson's method

The method of DAVIDSON [1975] is sometimes viewed as a generalized Arnoldi method, but other than the idea of working with a subspace, it has not much in common.

The main idea behind Davidson's method is the following one. Suppose we have some subspace $\mathcal{K}$ of dimension k, for which we have an orthonormal basis u_1, ..., u_k. If we denote U_k as the n by k matrix with columns u_i, then the matrix A reduced to this subspace is defined as $B_k = U_k^* A U_k$.

The eigenvalues of this matrix B_k are the Ritz values of A, with respect to the subspace, and an eigenvector s of B_k leads to a Ritz vector $y = U_k s$ of A. Indeed, let θ, s define an eigenpair of B_k:

$$B_k s = \theta s,$$

then we see that $\theta, U_k s$ satisfy a Ritz-Galerkin condition for the residual of this approximated eigenpair

$$A_k U_k s - \theta U_k s \perp \{u_1, u_2, \ldots, u_k\},$$

which is equivalent to $U_k^* A U_k s - \theta s = B_k s - \theta s = 0$.

Now we compute the residual for this approximated eigenpair

$$r := Ay - \theta y.$$

In view of the fast convergence of Shift-and-invert with Rayleigh quotients, Davidson suggested to focus on the residual and to try to enlarge the remaining component of the wanted eigenvector in the residual.

Davidson, a chemist, had diagonally dominant matrices in mind, and for a strongly diagonally dominant matrix we can approximate a shift-and-invert step by working with $(D_A - \theta I)^{-1}$ instead of $(A - \theta I)^{-1}$. This leads to a new vector $t = (D_A - \theta I)^{-1} r$, where D_A denotes the diagonal of A. This is the *expansion vector* for the current subspace: the vector t is orthonormalized with respect to U_k, and the resulting vector defines u_{k+1}.

It seems plausible that this will lead to a fast converging method if A is diagonally dominant, and, indeed, the method became quite popular for such problems that play an important problem, in particular in theoretical chemistry problems. This Davidson method is given in Figure 28.

```
u_1 = v/||v||_2
for k = 1, ..., m
    w_k = A u_k
    Compute k-th row and column of B_k
    for j = 1 : k - 1
        B(j,k) = u_j^* w_k
        B(k,j) = u_k^* w_j
    end
    B(k,k) = u_k^* w_k
    Compute largest eigenvalue θ of B_k and corresponding
    eigenvector s, with components s_(j)
    z = Σ_{j=1}^{k} s_(j) u_j
    r = Az - θz
    t = (D_A - θI)^{-1} r
    for j = 1, ..., k
        t = t - (u_j^* t) u_j
    end
    u_{k+1} = t/||t||_2
end
```

Figure 28: Davidson's Method with Diagonal Preconditioning

There is something very unpleasant in the heuristics behind this method. If we apply Davidson's method, by way of experiment, to an extremely diagonally dominant matrix, say a diagonal matrix, then this method stagnates. Indeed, we have that $r = (A - \theta I)y$, and hence $y = (A - \theta I)^{-1}r$, and since $y = U_k s$, this does not lead to an expansion for the current subspace.

This brings the method in a suspect position: the argument for faster convergence was to use a diagonal approximation for the diagonally dominant matrix A, but this is a fishy argument, because in the 'ideal' diagonal case, it will lead to failure. This helps to explain that numerical analysts felt not at ease with this method for a long time, and the unpleasant situation was more strongly felt because of the undeniable success of the method for relevant classes of matrices.

39 Numerical Example

In order to illustrate the behavior of the Davidson algorithm for increasingly better approximations of $(A - \theta I)^{-1}$, we consider the following tridiagonal matrix A of order 200.

The diagonal elements are defined as $a_{i,i} = 2.4 + i/2$, for $i = 1, \ldots, 199$, $a_{200,200} = 2.4 + 200/1.5$, and the codiagonal elements are given by $a_{i,i+1} = a_{i+1,i} = 1$. The starting vector v for the search space was chosen as $v_i = 0.03$, for $i = 1, \ldots, 199$, $v_{200} = 1$. The choice for this example may need some justification. We wish to demonstrate how Davidson's method suffers from too accurate preconditioning and we want to show later how this can be repaired by the so-called *Jacobi-Davidson* method.

We mimic the accuracy of preconditioning by solving the equation $(A - \theta I)y = r$ with different numbers of GMRES steps. For this purpose, we need a testing example with non-constant diagonal, otherwise the Krylov subspaces for different θ coincide (if the dimension is the same), and the expansion vector for the different techniques would be the same. Moreover, for a constant diagonal, the standard preconditioning in Davidson's method, that is $D_A - \theta I$, amounts to a scaling of the vector r and leads to r as expansion vector (in that case the method would be equivalent, in exact arithmetic, with Arnoldi's method).

In order to make GMRES effective for a small number of steps, we have chosen an example where the wanted eigenvalue is somehow isolated from the remaining spectrum in a relative sense. Finally, we have chosen the starting vector so that already in the first iteration of the Davidson method the value of θ is larger than the second largest eigenvalue of A. This helps to prevent delay of convergence in early phases of the iteration process.

Of course, such phenomenons may happen in reality and do happen indeed, but here we wanted to show the 'unperturbed' convergence behavior in good circumstances. In practical situations one can try to avoid initial misconvergence by selecting a proper shift strategy, for instance, by keeping a fixed target μ, instead of the recent θ, for a sufficient number of successive Davidson iterations.

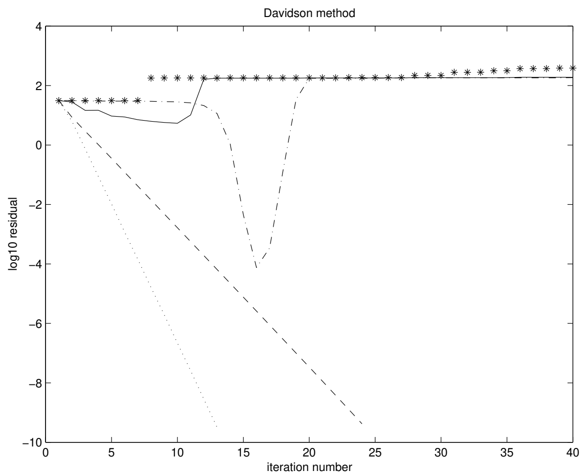

Figure 29: Residuals for Davidson's method for different scenarios

For five different situations, we have plotted the norms of the residuals for the largest Ritz pairs $\|(A-\theta I)z\|_2$, with $\|z\|_2 = 1$, computed with the Davidson method, in Fig. 29:

1. the solid line indicates the results for the standard preconditioner, that is, expanding the subspace with $(D_A - \theta I)^{-1} r$. We see that after some convergence in the first 10 iterations, there is no further convergence.

2. The dashed curve $(--)$ gives the results for the situation where the solution of $(A - \theta I)y = r$ is approximated with one step of unpreconditioned GMRES. In this case the method leads, in exact arithmetic, to the same Ritz approximations as Arnoldi's method. We get linear convergence and a reduction to 10^{-10} in about 25 Davidson iterations.

3. The dotted curve $(\cdots)$ represents the convergence history for two steps of GMRES. The convergence is now faster, as one might expect with a better solution of the expansion equation.

4. However, as argued in our discussion on the Davidson method, we would expect some form of stagnation if the expansion equation $(A - \theta I)y = r$ is solved too accurately, and one would expect to see some delaying effects already for moderate accuracies. Indeed, that is what we see, for 5 steps of GMRES, in the dashed-dotted curve $(-.-)$.

Here we see something puzzling: after some initial stagnation there is a phase of fast reduction of the residual until a norm of 10^{-4}, and then we see divergence again. We will shed some more light on this in the next example.

5. For 10 steps of GMRES the $*$ marks give the convergence history. As might have been anticipated, the convergence is now worse than in the previous situation and there is complete stagnation, with a strange small upward jump after 7 iterations (we will see what happened there, in the next example).

The negative effects of a too accurate solution of the expansion equation seem to be clearly visible from this example, but this is not the full story, as we will show now.

More on the numerical example One may expect that a quite accurate solution of the expansion equation $(A-\theta I)y = r$ returns a vector that makes a small angle with the current Ritz vector z. This may lead to inaccuracies in the orthogonalization procedure, and the deviation to z may hide valuable information for the unknown wanted eigenvector.

In this example we have done the same experiment as in the previous example, but now with repeated Gram-Schmidt for the orthogonalization of the expansion vector with respect to the existing basis vectors, collected in U. This leads to the convergence histories shown in Fig. 30.

We see that the situation has very much improved, with respect to the situation shown in Fig. 29. In particular, for diagonal preconditioning ($y = (D_A - \theta I)^{-1}r$) we see now fast convergence after about 17 steps, instead of divergence (the solid curve). Also the case for 5 steps of GMRES has improved. The initial fast convergence indicated after some 13 steps in Fig. 29, sets through (the dashed-dotted curve). However, the initial delay, for instance with respect to the dotted curve (with two steps of GMRES) has remained. Also, the apparantly too accurate solution with 10 steps of GMRES, could not be turned to a convergent iteration for Davidson's process (the $*$ marks), although the small upward jump after 7 steps has disappeared.

40 Davidson's method and preconditioning

The matrix $(D_A - \theta I)^{-1}$, or the way of solution of the expansion equation $(A - \theta I)y = r$, can be viewed as a preconditioner for the vector

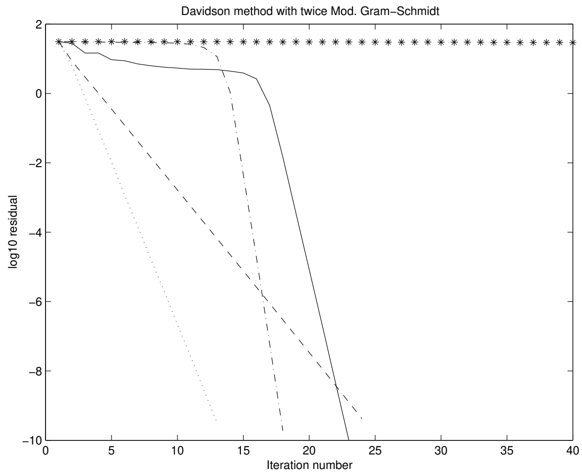

Figure 30: Davidson's method with repeated Gram-Schmidt

r. Since this is a diagonal preconditioner, it is sometimes referred to as Jacobi-preconditioning and this may lead to some confusion with the Jacobi-Davidson methods that we will discuss later.

The name Jacobi in the latter method refers to a different idea of Jacobi. Davidson [1993] suggests that his algorithm (more precisely: the Davidson-Liu variant of it) may be interpreted as a Newton-Raphson scheme, and this has been used as an argument to explain its fast convergence. We will come to this later; for the moment we leave it to the remark that it may be viewed as an inexact Newton method.

Of course, other preconditioners have been proposed for $A - \theta I$, see, for instance, Crouzeix, Philippe and Sadkane [1994], Morgan [1991], and Morgan and Scott [1986]. But, again, because $(A - \theta_k I)^{-1}$ would map r onto u_k, and would not lead to an expansion of the search space, the preconditioner should not be too good.

In this sense, the analysis in Crouzeix, Philippe and Sadkane [1994] is enlightening. In fact, their result shows that the preconditioner should not be exact in the direction of the approximated eigenvector to which θ corresponds. This underlines that approximating that operator does not seem to be the right viewpoint.

Originally, the Davidson method had been proposed for symmetric matrices, but in many of the cited publications it has been mentioned that the method can be used successfully for some unsymmetric matrices as well. In Figure 28, we have presented the algorithm for unsymmetric A; if A is symmetric then roughly half of the inner products for the construction of the then symmetric matrix B_k can be saved.

41 The methods of Jacobi

In his paper of 1846, JACOBI [1846] used rotations in order to make his system of linear equations, associated with eigenvalue computations, more diagonally dominant (see BODEWIG [1956, Chapter IV.3.1], DEN BOER [1991]; see also Chapter 6). After he had obtained a diagonally dominant system, he derived from this a linear system for the orthogonal complement of an approximate eigenvector, in his case a canonical unit vector. This linear system was then solved by Gauss-Jacobi iterations. In order to make our presentation more transparant, we will refer to the technique of constructing this linear system for the correction vector, as Jacobi's orthogonal complement correction (JOCC). We will explain this in more detail. Jacobi's method will make it easier for us to get more insight in the convergence of Davidson's method and it will lead us eventually to a class of iterative algorithms that does not suffer from the drawbacks that are inherent to Davidson's method.

The Orthogonal Complement Approach JACOBI [1845] considered an eigenvalue problem as a system of linear equations for which an iterative linear solver, the Jacobi or Gauss-Jacobi iteration (e.g., GOLUB and VAN LOAN [1996, Section 10.1.1]), might be applicable.

Suppose we have a diagonally dominant matrix A, of which $a_{1,1} = \alpha$ is the largest diagonal element. Then α is an approximation for the largest eigenvalue λ, and e_1 is an approximation for the corresponding eigenvector u. In modern matrix notation (which was unknown in Jacobi's time) his approach can be presented as follows. Consider the eigenvalue problem

$$A\begin{pmatrix} 1 \\ z \end{pmatrix} = \begin{pmatrix} \alpha & c^T \\ b & F \end{pmatrix}\begin{pmatrix} 1 \\ z \end{pmatrix} = \lambda\begin{pmatrix} 1 \\ z \end{pmatrix}, \qquad (41.1)$$

where F is a square matrix, α is a scalar, and b, c, and z are vectors of appropriate size. We are interested in the eigenvalue λ that is close in some sense to α, and in the corresponding eigenvector $u = (1, z^T)^T$, with component $(0, z^T)$ orthogonal to e_1. Problem (41.1) is equivalent with

$$\begin{aligned} \lambda &= \alpha + c^T z, & (41.2) \\ (F - \lambda I)z &= -b. & (41.3) \end{aligned}$$

Jacobi proposed to solve (41.3) iteratively with what is now known as Gauss-Jacobi iterations, with $z_1 = 0$, and an updated approximation for λ, using (41.2):

$$\left\{ \begin{array}{rcl} \theta_k & = & \alpha + c^T z_k \\ (D - \theta_k I) z_{k+1} & = & (D - F) z_k - b, \end{array} \right. \tag{41.4}$$

where D is the diagonal of F (although θ_k is not a Ritz value, we have used it to characterize it as an eigenvalue approximation).
Jacobi solved, as is also customary now, the update y_k for z_k from the diagonal system rather than z_{k+1} itself. Therefore, a better representation of the JOCC method would be

$$\left\{ \begin{array}{rcl} \mu_k & = & c^T y_k \\ \theta_{k+1} & = & \theta_k + \mu_k \\ z_{k+1} & = & z_k + y_k \\ (D - \theta_{k+1} I) y_{k+1} & = & (D - F) y_k + \mu_k z_{k+1} \end{array} \right. \tag{41.5}$$

with $z_1 = 0$, $\theta_1 = \alpha$ and $y_1 = -(D - \alpha I)^{-1} b$. However, in connection with Davidson's method, representation (41.4) simplifies the discussion.

Short discussion on the JOCC method Jacobi was well aware of the fact that the Gauss-Jacobi iteration converges (fast) if the matrix is (strongly) diagonally dominant. Therefore, he proposed to perform a number of steps of the Jacobi diagonalization method in order to obtain a (strongly) diagonally dominant matrix, before applying the JOCC method.

Since Jacobi was interested in the eigenvalue closest to α, he did enough steps with the diagonalization method in order to obtain a diagonally dominant $F - \alpha I$, and so that $F - \theta_k I$ was also diagonally dominant. This can be done provided that λ is a simple eigenvalue of A.

The application of the diagonalization method can be viewed as a technique to improve the initial guess e_1, i.e., the given matrix is rotated so that e_1 is closer to the (rotated) eigenvector u. These rotations were only done at the start of the iteration process, and this process was carried out with fixed F and D.

However, note that in Jacobi's approach we are looking, at all stages, for the orthogonal complement to the *initial* approximation $u_1 = e_1$. We do not take into account that better approximations $u_k = (1, z_k^T)^T$ become available in the process, and that it may be more efficient to try to compute the orthogonal complement $u - (u^T u_k) u_k$.

In the JOCC framework an improved approximation would have led to a similar situation as in (41.1), if we would have applied plane rotations on A, such that u_k would have been rotated to e_1 by these rotations. Therefore, what Jacobi did in the first step of the process only, could have been done *at each step* of the iteration process.

This is an exciting idea, since in Jacobi's approach the rotations were motivated by the desire to obtain more diagonally dominance, whereas our discussion suggests that one might take advantage of the information in the iteration process. Of course, this would have led to a different operator F in each step and this is an important observation for the formulation of our new algorithm.

42 The Jacobi-Davidson iteration method

From now on we will allow the matrix A to be complex, and in order to express this we use the notation v^* for the complex conjugate of a vector (if complex), or the transpose (if real), and likewise for matrices.

As we have stressed in the previous section, the JOCC method and Davidson's method can be seen as methods that attempt to find the correction to some initially given eigenvector approximation. In fact what we want is to find the orthogonal complement for our current approximation u_k with respect to the desired eigenvector u of A.

Jacobi suggested to compute the component of the wanted eigenvector x, orthogonal to the starting vector e_1, and for that purpose he considered a system that was obtained from the given system by skipping the first column and first row. In modern matrix terminology: he considered the projection of the given eigenproblem onto the orthogonal complement of e_1. Since we are interested in the component of x orthogonal to the most recent approximation u_k for x, we restrict ourselves to that component for the subspace $u_k^\perp$.

The orthogonal projection of A onto the subspace $u_k^\perp$ is given by

$$B = (I - u_k u_k^*) A (I - u_k u_k^*)$$

(we assume that u_k has been normalized). Because θ_k, the Ritz value, satisfies $\theta_k = u_k^* A u_k$, it follows that

$$A = B + A u_k u_k^* + u_k u_k^* A - \theta_k u_k u_k^*. \tag{42.1}$$

Suppose that we want to compute the eigenpair λ, x of A, for a λ close to θ_k, and u_k close to x. Then clearly the orthogonal component $v \perp u_k$, for which $x = u_k + v$, satisfies

$$A(u_k + v) = \lambda(u_k + v),$$

or, after inserting (42.1), and realizing that $Bu_k = 0$:

$$(B - \lambda I)v = -r_k + (\lambda - \theta_k - u_k^* Av)u_k, \tag{42.2}$$

where $r_k = Au_k - \theta_k u_k$.
Since the left-hand side and r_k are orthogonal to u_k, it follows that the factor for u_k must vanish, and hence v should satisfy:

$$(B - \lambda I)v = -r_k = (A - \theta_k I)u_k. \tag{42.3}$$

Since we do not know λ, we replace it, just as in JOCC and in Davidson's method, by the current approximation θ_k.[2] This leads to the correction equation

$$(I - u_k u_k^*)(A - \theta_k I)(I - u_k u_k^*)v = -r_k. \tag{42.4}$$

For the expansion of the subspace we solve equation (42.4), for a given Ritz value θ_k, and we expand the subspace with v. We compute a new Ritz pair with respect to the expanded subspace and we repeat the above procedure. This is the basis for the *Jacobi-Davidson Method*, proposed in Sleijpen and van der Vorst [1996a].

Let us consider the equation (42.4) in some more detail. Since $v \in u_k^\perp$, we have that $(I - u_k u_k^*)v = v$, and obviously equation (42.4) can be written as

$$(A - \theta_k I)v = -r_k + \alpha u_k, \tag{42.5}$$

with $\alpha = u_k^*(A - \theta_k I)v$. Of course, we do not know v, so we do not know α but this can be circumvented as follows. Let K be an approximation for $A - \theta_k I$, then we solve v from

$$Kv = -r_k + \alpha u_k.$$

The value for α follows from the orthogonality constraint for v:

$$\alpha = \frac{u_k^* K^{-1} r_k}{u_k^* K^{-1} u_k}.$$

This means that we have to carry out a solution step with K twice, which makes the method potentially expensive, but, as we will see later, this can be reduced to almost one action with K^{-1} if we solve (42.4) with an iterative solver with preconditioner K.

[2] If θ is not close to the wanted eigenvalue then it is often more efficient to replace it by a fixed target value that is close to the wanted eigenvalue. This helps to prevent convergence of θ to an undesired eigenvalue, or local stagnation of the convergence process.

If we take $K = A - \theta_k I$, then we get formally

$$\begin{aligned} v &= -(A-\theta_k I)^{-1} r + \alpha (A-\theta_k I)^{-1} u_k \\ &= -u_k + \alpha (A-\theta_k I)^{-1} u_k. \end{aligned}$$

Because u_k is already in our search subspace spanned by the columns of U_k, the subspace is effectively expanded by the normalized vector $(A - \theta_k I)^{-1} u_k$. This is the same vector that is generated in the Rayleigh-Quotient Iteration (RQI), so that in the case of exact solution of the correction method, the Jacobi-Davidson method reduces to an accelerated Rayleigh-Quotient iteration for the shifted and inverted matrix.

This method has cubic convergence if A is symmetric and quadratic convergence if A is unsymmetric (Ostrowski [1958], Ostrowski [1959]). The quadratic convergence of the Jacobi-Davidson method can also be concluded from the observation that the method can be derived as a Newton iteration process (see Sleijpen and van der Vorst [1996b]). Because of the fast convergence of the RQI method, we may not expect much by this acceleration. The acceleration will be more effective is the correction equation is solved only approximately, but note that we loose cubic or quadratic convergence in that case (just as in inexact Newton methods).

The Jacobi-Davidson method is based on a combination of two basic principles. The first one is to apply a Ritz-Galerkin approach for the eigenproblem $Ax = \lambda x$, with respect to some given subspace spanned by an orthonormal basis $v_1, \ldots, v_k$. The Ritz-Galerkin condition is

$$AV_k s - \theta V_k s \perp \{v_1, \ldots, v_k\},$$

which leads to the reduced system

$$V_k^* A V_k s - \theta s = 0, \tag{42.6}$$

where V_k denotes the matrix with columns v_1 to v_k. This equation has k solutions $(\theta_j^{(k)}, s_j^{(k)})$. The k pairs $(\theta_j^{(k)}, u_j^{(k)} := V_k s_j^{(k)})$ are the Ritz values and Ritz vectors of A with respect to the subspace spanned by the columns of V_k.

The other principle is to compute an approximation from (42.4) for the orthogonal complement of the current Ritz vector.

The resulting algorithm may be viewed as a combination of Jacobi's approach to look for the orthogonal complement of a given eigenvector approximation, and the Davidson algorithm for expanding the subspace in which the eigenvector approximations are constructed. This explains the origin of the name *Jacobi-Davidson* for this method. Note that we are free to use any method for the (approximate) solution of (42.4) and that it is not necessary to require diagonally dominance of A.

We will discuss now what happens if (42.4) is solved approximately. This question has been addressed in SLEIJPEN and VAN DER VORST [1996a], and has led to the following observations:

1. If we take the very crude approximation $v = -r_k$, then the method becomes mathematically equivalent with the Arnoldi method (and with Lanczos if A is symmetric), in the sense that, in exact arithmetic, the same subspaces are generated. The Jacobi-Davidson approach is not attractive in this case, since it does not detect that the basis can be generated in such a way that M is upper Hessenberg (or tridiagonal). However, this might be an advantage for severely ill-conditioned systems.

2. If we approximate the projected operator by $D_A - \theta_k I$, and skip the condition that $v \perp u_k$ then we obtain Davidson's method. Other suggestions for the approximation of $A - \theta_k I$, made in CROUZEIX, PHILIPPE and SADKANE [1994], MORGAN [1991], MORGAN [1992], MORGAN and SCOTT [1993], and DAVIDSON [1993], come down to better approximations for the inverse of $A - \theta_k I$, e.g., incomplete decompositions for this operator. However, as is well-known, this is a risky approach if we skip the Jacobi-Davidson condition $v \perp u_k$ (see SAAD [1992], CROUZEIX, PHILIPPE and SADKANE [1994]), since the exact inverse of this operator may lead to stagnation of the method, and therefore the approximation should not be too accurate (SAAD [1992]).

3. If we take full account of the restriction to the subspace orthogonal to u_k the we obtain the Jacobi-Davidson methods.

43 The iterative solution of the correction equation

In SLEIJPEN and VAN DER VORST [1996a], it is suggested to solve equation (42.4) only approximately, for instance by some steps of MINRES (PAIGE and SAUNDERS [1975]), when A is symmetric, or with GMRES (SAAD and SCHULTZ [1986], for unsymmetric A, with an appropriate preconditioner K

for $A - \theta_j^{(k)} I$. Because the residual depends on k and j, we will denote the solution of (42.4) by $v_j^{(k)}$.

Any approximation technique for $v_j^{(k)}$ is formally allowed, provided that the projectors $\left(I - u_j^{(k)} u_j^{(k)^*}\right)$ are taken into account.

We will now discuss how to use preconditioning for an iterative solver like GMRES or CGS, applied with equation (42.4). Suppose that we have a preconditioner K for the matrix $A - \theta_j^{(k)} I$, for which in some sense $K^{-1}(A - \theta_j^{(k)} I) \approx I$. Since $\theta_j^{(k)}$ varies with the iteration count k, so may the preconditioner, although it is often practical to work with the same K for different values of θ. Of course, the preconditioner K has to be restricted to the subspace orthogonal to $u_j^{(k)}$ as well, which means that we have to work effectively with

$$\widetilde{K} := \left(I - u_j^{(k)} u_j^{(k)^*}\right) K \left(I - u_j^{(k)} u_j^{(k)^*}\right).$$

At first sight this seems to involve a lot of overhead because of the projections $\left(I - u_j^{(k)} u_j^{(k)^*}\right)$ that encircle the matrix vector operations. But it all amounts to a handful of rather simple operations, as we will show now.

We will from now on assume that we use a Krylov solver with initial guess $t_0 = 0$ and with left preconditioning for the approximate solution of the correction equation (42.4). Since the starting vector is in the subspace orthogonal to $u_j^{(k)}$, all iteration vectors for the Krylov solver will be in that space. In that subspace we have to compute the vector $z := \widetilde{K}^{-1} \widetilde{A} v$, for a vector v supplied by the Krylov solver, and

$$\widetilde{A} := \left(I - u_j^{(k)} u_j^{(k)^*}\right) (A - \theta_j^{(k)} I) \left(I - u_j^{(k)} u_j^{(k)^*}\right).$$

We will do this in two steps and first we have to compute

$$\begin{aligned}
\widetilde{A} v &= \left(I - u_j^{(k)} u_j^{(k)^*}\right) (A - \theta_j^{(k)} I) \left(I - u_j^{(k)} u_j^{(k)^*}\right) v \\
&= \left(I - u_j^{(k)} u_j^{(k)^*}\right) y \\
\text{with } y &:= (A - \theta_j^{(k)} I) v \quad \text{since} \quad u_j^{(k)^*} v = 0\,.
\end{aligned}$$

Then, with left-preconditioning we have to solve $z \perp u_j^{(k)}$ from

$$\widetilde{K} z = \left(I - u_j^{(k)} u_j^{(k)^*}\right) y.$$

Since $z \perp u_j^{(k)}$, it follows that z satisfies

$$Kz = y - \alpha u_j^{(k)} \quad \text{or} \quad z = K^{-1}y - \alpha K^{-1}u_j^{(k)}.$$

The condition $z \perp u_j^{(k)}$, leads to

$$\alpha = \frac{u_j^{(k)^*} K^{-1} y}{u_j^{(k)^*} K^{-1} u_j^{(k)}}.$$

The vector $\widehat{y} := K^{-1}y$ is solved from $K\widehat{y} = y$ and, likewise, $\widehat{u} := K^{-1}u_j^{(k)}$ is solved from $K\widehat{u} = u_j^{(k)}$. Note that the last equation has to be solved only once in an iteration process for equation (42.4), so that effectively $i_S + 1$ operations with the preconditioner are required for i_S iterations of the linear solver.

Note also that one matrix vector multiplication with $\widetilde{K}^{-1}\widetilde{A}$, in an iteration of the Krylov solver, requires only one inner product and one vector update instead of the four actions with the projector $\left(I - u_j^{(k)} u_j^{(k)^*}\right)$. The solution of (42.4), by a Krylov subspace method with preconditioning, has been worked out in Fig. 31.

The algorithms for preconditioning of the correction equation have been described previously in Sleijpen, van der Vorst and Meijerink [1998].

If we form an approximation for $v_j^{(k)}$ in (42.4) as

$$\widetilde{v}^{(k)} := K^{-1}r_k - \alpha K^{-1}u_j^{(k)},$$

with α such that

$$\widetilde{v}^{(k)} \perp u_j^{(k)},$$

and without acceleration by an iterative solver, then we obtain a process which was suggested in Olsen, Jørgensen and Simons [1990].

44 A numerical example

We will now illustrate by a numerical example what we may observe as diffences between the Davidson method and the Jacobi-Davidson method. For this purpose we return to the same example as discussed in Section 39, with the same starting vector v.

Again, we consider 5 different situations: diagonal scaling (as in the standard Davidson method) and solutions of the Davidson expansion equation and the Jacobi-Davidson correction equation with (unpreconditioned)

Preconditioner $\widetilde{K} := (I - uu^*)K(I - uu^*)$
for $\widetilde{A} := (I - uu^*)(A - \theta I)(I - uu^*)$:

Solve $\widehat{u}$ from $K\widehat{u} = u$, $\mu := u^*\widehat{u}$
Compute $\widetilde{r} := \widetilde{K}^{-1}r$ as:
(b′) solve $\widehat{r}$ from $K\widehat{r} = r$
(c′) $\widetilde{r} = \widehat{r} - \frac{u^*\widehat{r}}{\mu}\,\widehat{u}$
Apply Krylov subspace method with start $t_0 = 0$,
with operator $\widetilde{K}^{-1}\widetilde{A}$, and right-hand side $-\widetilde{r}$.
$z = \widetilde{K}^{-1}\widetilde{A}v$ for given v is computed as:
(a) $y = (A - \theta I)v$
(b) *solve* $\widehat{y}$ from $K\widehat{y} = y$
(c) $z = \widehat{y} - \frac{u^*\widehat{y}}{\mu}\,\widehat{u}$

Figure 31: Iterative solution of the correction equation with left-preconditioning

GMRES. In Fig. 30, we have shown the convergence histories, represented by the norm of the residual for the selected Ritz pair, for the algorithm with repeated modified Gram-Schmidt.

In brief: the solid line gives standard diagonal preconditioning, the dashed line represents the results for one step of GMRES, the dotted line represents the results for 2 steps of GMRES, the dashhed-dotted curve for 5 steps of GMRES, and the $*$ marks for 10 steps of GMRES.

In Fig. 32, we have collected the results for the Jacobi-Davidson method for the same situations. We see striking differences. First, the diagonal preconditioner together with the projection in order to get a correction in the co-space of the current Ritz vector leads to a faster convergent process, about 8 iterations instead of the 23 Davidson iterations. One single unpreconditioned GMRES step leads to a process that, in exact arithmetic, is equivalent with Arnoldi's method and it is not surprising that the results are the same as for Davidson.

Also the results for 2 steps GMRES are about the same. The major differences, as expected, arise when we solve the correction equation more accurately, and this is seen by the dashed-dotted curves (5 steps GMRES)

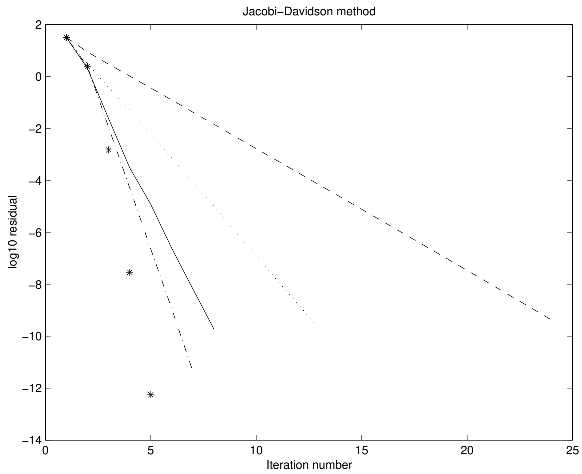

Figure 32: Residuals for Jacobi-Davidson for different scenarios

and most spectacular, of course, by the $*$ marks (10 steps of GMRES).

45 A simple Jacobi-Davidson Algorithm

The basic form of the Jacobi-Davidson algorithm is given in Fig. 33. In each iteration of this algorithm an approximated eigenpair (θ, u) for the eigenpair of the matrix A, corresponding to the largest eigenvalue (in absolute value) of A, is computed. The iteration process is terminated as soon as the norm of the residual $Au - \theta u$ is below a given threshold ϵ. One has to be careful with the target θ_k, as has been argued in Fokkema, Sleijpen and van der Vorst [1998]. If the initial approximation v_0 for the wanted eigenvector makes a significant angle with that eigenvector then the Ritz value θ of A with respect to v_0 may be closer to some other eigenvalue of A than the wanted one. Since we are actually carrying out a Newton iteration for a solution of $Ax - \lambda x = 0$, for $\|x\|_2 = 1$, the convergence may initially set out for the closest by eigenvalue.

Only after the subspace becomes large enough, and because of the inexact solutions a component of the wanted eigenvector has entered the subspace, the selected Ritz value will move away from the initially closest by eigenvalue. It may then move to some other eigenvalue and the history may repeat itself a few times before eventually the convergence to the wanted largest eigenvalue sets in.

These phases of misconvergence may be avoided by selecting a target τ instead of θ_k in the correction equation. This target should ideally have the property that it is closer to the largest eigenvalue than to any other eigenvalue. Only when the Ritz values seem to converge to the largest one, the target value is replaced by the current Ritz value in the correction equation.

```
        Start with t = v_0, starting guess
        for m = 1, ...
(1)         for i = 1, .., m - 1
                t = t - (v_i^* t) v_i
            v_m = t/||t||_2, v_m^A = A v_m
(2)         for i = 1, .., m - 1
                M_{i,m} = v_i^* v_m^A
                M_{m,i} = v_m^* v_i^A
            M_{m,m} = v_m^* v_m^A
(3)         Compute the largest eigenpair Ms = θs
            of the m by m matrix M, (||s||_2 = 1)
            u = Vs with V = [v_1, ..., v_m]
            u^A = V^A s with V^A = [v_1^A, ..., v_m^A]
            r = u^A - θ u
(4)         if (||r||_2 ≤ ε), λ~ = θ, x~ = u, then STOP.
(5)         Solve (approximately) a t ⊥ u from
                (I - uu^*)(A - θI)(I - uu^*)t = -r
```

Figure 33: Jacobi-Davidson Algorithm for $\lambda_{\max}(A)$

To apply this algorithm we need to specify a starting vector v_0, and a tolerance ϵ.

On completion an approximation for the largest eigenvalue (in absolute value) $\lambda = \lambda_{\max}(A)$ and its corresponding eigenvector $x = x_{\max}$ is delivered. The computed eigenpair $(\widetilde{\lambda}, \widetilde{x})$ satisfies $\|A\widetilde{x} - \widetilde{\lambda}\widetilde{x}\| \le \epsilon$.

We will now describe some implementation details, referring to the respective phases in Fig. 33.

(1) This is a modified Gram-Schmidt process for the orthogonalization of

the new vector t with respect to the set $v_1, \ldots, v_{m-1}$. If $m = 1$ then this is an empty loop. Let t_{in} represent the vector before the start of modified Gram-Schmidt, and t_{out} the result before normalisation. It is advisable (see DANIEL *et al* [1976]) to repeat the Gram-Schmidt process one time if $\|t_{out}\|_2/\|t_{in}\|_2 \leq \kappa$, where κ is a modest constant, say $\kappa = .25$ (see also our discussion on the stable computation of a basis for the Krylov subspace in Section 14).

This guarantees that the loss of orthogonality of v_m, with respect to v_1, ..., v_{m-1}, is restricted to $1/\kappa$ times machine precision, in a relative sense.More precisely, with $E_m := V_m^* V_m - I$, we have that

$$\|E_{m+1}\|_2 \leq \frac{1}{\kappa}\|E_m\|_2.$$

The modified Gram-Schmidt loop in (1) can then be replaced by orthogonalization with iterative refinement as displayed in Alg. 34.
This modified Gram-Schmidt may be replaced by classical Gram-Schmidt (with iterative refinement), which has the advantage that this can be carried out with matrix vector operations.

> select a $\kappa < 1$, say $\kappa = .25$
> $\tau_{in} = \|t\|_2$
> **for** $i = 1, .., m-1$
> $t = t - (v_i^* t) v_i$
> **if** $\|t\|_2/\tau_{in} \leq \kappa$
>   **for** $i = 1, .., m-1$
> $t = t - (v_i^* t) v_i$

Figure 34: Modified Gram-Schmidt with one step of iterative refinement

(2) Computation of the mth column and mth row of the matrix $M := V^*AV$. The matrix V denotes the n by m matrix with columns v_j, V^A is the matrix with columns v_j^A. If A is Hermitian, then M is also Hermitian and in that case almost half of the work in the construction of M can be saved.

(3) The computation of the eigenpair with largest eigenvalue (in absolute value) of the $m \times m$ matrix M can be done with the appropriate

routines from LAPACK (ANNDERSON *et al* [1992]) or other software collections.

The vector u^A may be alternatively computed as $u^A = Au$, depending on which way is cheapest. The choice is between computing Au and $V^A s$; if A has less than m nonzero elements on average per row, then the computation via Au is preferable. If u^A is computed as Au then it is not necessary to store the vectors v_j^A.

(4) The algorithm is terminated if $\|Au - \theta u\|_2 \leq \epsilon$. If A is Hermitian then it has an eigenvalue λ for which $|\lambda - \theta| \leq \epsilon$. For the corresponding normalized eigenvector there is a similar bound on the angle, provided that λ is simple and well separated from the other eigenvalues of A, see Section 5. That case leads also to a sharper bound for λ.

(5) The approximate solution for the expansion vector t can be computed with a Krylov solver, for instance, Bi-CGSTAB or GMRES. Note that if A is unsymmetric then the eigenvalues and eigenvectors may be complex. This implies that the correction equation is complex and we have to take this into account for the iterative solver. For iterative Krylov subspace methods see BARRETT *et al* [1994].

An algorithm for the approximate solution, with a left-preconditioned Krylov subspace method of choice, is given in Alg. 31.

The approximate solution has to be orthogonal to u, but that is automatically the case with Krylov solvers if one starts with an initial guess orthogonal to u, for instance $t_0 = 0$. In most cases it is not necessary to solve the correction equation to high precision; a relative precision of 2^{-m} in the mth iteration seems to suffice. This strategy has been proposed for inexact Newton methods and it is natural to adopt it for Jacobi-Davidson, which can be interpreted as an (inexact) Newton method. It is advisable to put a limit to the number of iteration steps for the iterative solver.

In order to restrict storage, the algorithm can be terminated at some appropriate value $m = m_{\max}$, and restarted with for v_0 the latest vector u. We will describe a variant of the Jacobi-Davidson algorithm with a more sophisticated restart strategy in Section 46.

46 Restart and deflation strategies

Restart strategy for one single eigenpair The increasing storage or the computational overhead, for increasing dimension m of the subspace, may make it necessary to restart. An obvious way to restart is to take the most recent approximation for the desired eigenvector. However, this may not be the most efficient strategy for restarting. With any restart by a single vector we discard possibly valuable information that is contained in the remaining part of the subspace. Unless we have an invariant subspace, all vectors in the subspace contain information for the wanted eigenvector.

After restart with one single vector we see the effect of the lost information by a slowdown in the speed of convergence. Therefore it is often better to restart with a set of vectors representing a subspace that contains more information for the wanted eigenpair and a good strategy is to restart with the subspace spanned by the Ritz vectors of a small number of the Ritz values closest to a specified target value.

Computation of several eigenpairs In some circumstances the Jacobi-Davidson method has apparent disadvantages with respect to the standard Arnoldi method. 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 of Ritz values towards a particular eigenvalue goes hand in hand with simultaneous convergence of other Ritz values towards other eigenvalues. So whereas Jacobi-Davidson deliberately aims for one particular eigenvalue (closest to some target), the Arnoldi process leads to approximations for (eventually) all eigenvalues, and in the early phases of the iteration it leads usually to approximations for isolated exterior eigenvalues.

For Jacobi-Davidson the obvious approach would be to restart with a differently selected Ritz pair, with no guarantee 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. This is no problem for Hermitian matrices A, because they have an orthonormal eigenvector system. Since we want to avoid to work with non-orthogonal reduction or deflation operators, we propose to work with *Schur vectors* for non-Hermitian matrices. These Schur vectors are defined as the columns of the orthogonal matrix Q that transforms A to upper triangular form:

$$Q^*AQ = R.$$

The eigenvalues of A appear along the diagonal of R and the eigenvectors x of A can be computed from the eigenvectors y of R as $x = Qy$.

In FOKKEMA, SLEIJPEN and VAN DER VORST [1998], an algorithm is proposed by which several eigenpairs can be computed. The algorithm is based on the computation of a partial Schur form of A:

$$AQ_k = Q_k R_k,$$

where Q_k is an $n \times k$ orthonormal matrix, and R_k is a $k \times k$ upper triangular matrix, with $k \ll n$. Note that if (x, λ) is an eigenpair of R_k, then $(Q_k x, \lambda)$ is an eigenpair of A.

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

Step I: Given an orthonormal subspace basis $v_1, \ldots, v_i$, with matrix V_i, compute the projected matrix $M = V_i^* A V_i$. For the $i \times i$ matrix M we compute the complete Schur form $MU = US$, with $U^*U = I$, and S upper triangular. This can be done with the standard QR algorithm (see Section 10).
Then we order S such that the $|s_{i,i} - \tau|$ form a nondecreasing row for increasing i. The first few diagonal elements of S 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. This reordering can be done while preserving the upper triangular structure of S, for details see FOKKEMA [1996].

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: $Aq = \lambda q$. The question is how to expand this partial Schur form of dimension 1. This will be shown in step **II**.

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

$$A\,[Q_k, q] = [Q_k, q] \begin{bmatrix} R_k & s \\ & \lambda \end{bmatrix}$$

with $Q_k^* q = 0$.
After some standard linear algebra manipulations it follows that

$$(I - Q_k Q_k^*)(A - \lambda I)(I - Q_k Q_k^*) q = 0,$$

which expresses that the new pair (q, λ) is an eigenpair of

$$\widetilde{A} = (I - Q_k Q_k^*) A (I - Q_k Q_k^*).$$

This pair can be computed by applying the Jacobi-Davidson algorithm (with Schur form reduction, as in step **I** for $\widetilde{A}$).

Although we see that after each converged eigenpair the explicitly deflated matrix $\widetilde{A}$ leads to more expensive computations, it is shown in Fokkema, Sleijpen and van der Vorst [1998], by numerical experiments, that the entire procedure may lead to a very efficient computational process. An explanation for this is that after convergence of some eigenvectors, corresponding to eigenvalues closest to the target, the deflated matrix $\widetilde{A}$ will be better conditioned. This means that the correction equation in the Jacobi-Davidson step is more easily solved by an iterative solver.

The correction equation may be solved by a preconditioned iterative solver, and it is shown in Fokkema, Sleijpen and van der Vorst [1998], that the same preconditioner can be used for different eigenpairs. Hence, it pays to construct better preconditioners.

With the proposed restart strategy, in Section 46, it is to be expected that when a Ritz value is close enough to an eigenvalue, the remaining part of the current subspace will already have rich components in nearby eigenpairs, since we have selected in all steps the Ritz vectors for Ritz values close to the desired eigenvalue. We can use this information as the basis for a subspace for the computation of a next eigenvector, after the eigenvalue equation has been properly deflated in order to prevent the just found eigenvector to re-enter the computational process.

Let $q_1, \ldots, q_{k-1}$ denote the accepted Schur vector approximations and let us assume that these vectors are orthonormal in good precision. Deflation with approximate Schur vectors, with residuals $\|Aq_j - QRe_j\|_2$ in the order of ϵ, may introduce an error of order ϵ^2 in the eigenvalues, provided that the computed eigenvalues are well separated from the remaining ones (Parlett [1980, Section 5.1]).

The matrix Q_{k-1} has the vectors q_j as its columns. In order to find the next Schur vector q_k, we apply the Jacobi-Davidson algorithm to the deflated matrix

$$(I - Q_{k-1} Q_{k-1}^*)\, A\, (I - Q_{k-1} Q_{k-1}^*),$$

and this leads to a correction equation of the form

$$P_m (I - Q_{k-1} Q_{k-1}^*)\, (A - \theta_j^{(m)} I)\, (I - Q_{k-1} Q_{k-1}^*) P_m t_j^{(m)} = -r_j^{(m)}, \quad (46.1)$$

with $P_m := \left(I - u_j^{(m)} u_j^{(m)^*}\right)$, that has to be solved for the correction $t_j^{(m)}$ to each new Schur vector approximation $u_j^{(m)}$, with corresponding Ritz value $\theta_j^{(m)}$.

In Fokkema, Sleijpen and van der Vorst [1998], it is shown, by numerical evidence, that the explicit deflation in the correction equation (46.1) against the vectors represented by Q_{k-1} is highly recommended for the correction equation, but is is not necessary to include this deflation in the computation of the projected matrix (the projection of A onto the subspace spanned by the successive approximated solutions v_j of the correction equation, in the search for the kth Schur vector). The projected matrix can be computed as $V_m^* A V_m$, without significant loss of accuracy.

47 Preconditioning

Preconditioning for an iterative solver like GMRES or CGS, applied with equation (46.1) is only slightly more complicated than in the single vector case (cf. Sec. 43). Suppose that we have a left preconditioner K available for the operator $A - \theta_j^{(m)} I$. Let $\widetilde{Q}$ denote the matrix Q_{k-1} expanded with $u_j^{(m)}$ as its kth column. In this case, the preconditioner K has to be restricted to the subspace orthogonal to $\widetilde{Q}$ as well, which means that we have to work effectively with

$$\widetilde{K} := (I - \widetilde{Q}\widetilde{Q}^*) K (I - \widetilde{Q}\widetilde{Q}^*).$$

Similar to the single vector case this can be realized in a surprisingly efficient way.

Assume that we use a Krylov solver with initial guess $t_0 = 0$ and with left preconditioning for the approximate solution of the correction equation (46.1). Since the starting vector is in the subspace orthogonal to $\widetilde{Q}$, all iteration vectors for the Krylov solver will be in that space. In that subspace we have to compute the vector $z := \widetilde{K}^{-1} \widetilde{A} v$, for a vector v supplied by the Krylov solver, and

$$\widetilde{A} := (I - \widetilde{Q}\widetilde{Q}^*)(A - \theta_j^{(m)} I)(I - \widetilde{Q}\widetilde{Q}^*).$$

This is done in two steps and first we compute

$$\begin{aligned} \widetilde{A} v &= (I - \widetilde{Q}\widetilde{Q}^*)(A - \theta_j^{(m)} I)(I - \widetilde{Q}\widetilde{Q}^*) v \\ &= (I - \widetilde{Q}\widetilde{Q}^*) y \\ \text{with } \; y &:= (A - \theta_j^{(m)} I) v \;\; \text{since} \;\; \widetilde{Q}^* v = 0 \,. \end{aligned}$$

Then, with left-preconditioning we have to solve $z \perp \widetilde{Q}$ from

$$\widetilde{K} z = (I - \widetilde{Q}\widetilde{Q}^*) y.$$

Since $\widetilde{Q}^* z = 0$, it follows that z satisfies $Kz = y - \widetilde{Q}\vec{\alpha}$ or $z = K^{-1}y - K^{-1}\widetilde{Q}\vec{\alpha}$. The condition $\widetilde{Q}^* z = 0$ leads to

$$\vec{\alpha} = (\widetilde{Q}^* K^{-1} \widetilde{Q})^{-1} \widetilde{Q}^* K^{-1} y.$$

The vector $\widehat{y} := K^{-1}y$ is solved from $K\widehat{y} = y$ and, likewise, $\widehat{Q} := K^{-1}\widetilde{Q}$ is solved from $K\widehat{Q} = \widetilde{Q}$.

Note that the last set of equations has to be solved only once in an iteration process for equation (46.1), so that effectively $i_S + k$ operations with the preconditioner are required for i_S iterations of the linear solver. The matrix vector multiplication with the left-preconditioned operator, in an iteration of the Krylov solver, requires only one operation with $\widetilde{Q}^*$ and $K^{-1}\widetilde{Q}$, instead of the four actions of the projector operator $(I - \widetilde{Q}\widetilde{Q}^*)$. This is similar to the situation for the single vector case, as described in Section 43.

Obvious savings can be realized if the operator K is kept the same for a number of successive eigenvalue computations (for details, see SLEIJPEN, VAN DER VORST and MEIJERINK [1998]). In that case columns of $\widehat{Q}$ can be saved from previous steps.

48 A complete algorithm

A scheme that leads to the computation of a partial Schur form

$$AQ_k = Q_k R_k,$$

is given in Alg. 35. The algorithm represented there is referred to in literature as JDQR, see FOKKEMA, SLEIJPEN and VAN DER VORST [1998]. This scheme includes the possibility for restart when the dimension of the current subspace exceeds $m_{\max}$.

The scheme computes $k_{\max}$ eigenvalues close to a target τ in the complex plane. Here we have to be necessarily unprecise, since the eigenvalues of a general non-Hermitian matrix are not ordered in the complex plane. Which Ritz values are delivered as close eigenvalues depends amongst others on the angles of the corresponding eigenvectors.

Usually, the selection from Ritz values is appropriate if τ is chosen somewhere at the exterior of the distribution of eigenvalues. If we want eigenvalues of A close to some interior point then the scheme may be much less satisfactory, and we propose to work in such cases with harmonic Ritz values. This situation is discussed in Section 49.

$t = v_0$, $k = 0$, $m = 0$; $Q = [\,]$, $R = [\,]$
while $k < k_{\max}$
(1) **for** $i = 1, .., m$
$t = t - (v_i^* t) v_i$
$m = m + 1$, $v_m = t / \|t\|_2$, $v_m^A = A v_m$
(2) **for** $i = 1, .., m - 1$
$M_{i,m} = v_i^* v_m^A$, $M_{m,i} = v_m^* v_i^A$
$M_{m,m} = v_m^* v_m^A$
(3) Make a Schur decomposition $M = STS^*$,
S unitary and T upper triangular,
such that: $|T_{i,i} - \tau| \leq |T_{i+1,i+1} - \tau|$
$u = V s_1$, $u^A = V^A s_1$, $\theta = T_{1,1}$, $r = u^A - \theta u$
$\widetilde{a} = Q^* r$, $\widetilde{r} = r - Q\widetilde{a}$
(4) **while** $\|\widetilde{r}\|_2 \leq \epsilon$
$R = \begin{pmatrix} R & \widetilde{a} \\ 0 & \theta \end{pmatrix}$, $Q = [Q, u]$, $k = k + 1$
if $k = k_{\max}$ **then** STOP
$m = m - 1$
(5) **for** $i = 1, .., m$
$v_i = V s_{i+1}$, $v_i^A = V^A s_{i+1}$, $s_i = e_i$
$M =$ lower m by m block of T
$u = v_1$, $\theta = M_{1,1}$, $r = v_1^A - \theta u$, $\widetilde{a} = Q^* r$, $\widetilde{r} = r - Q\widetilde{a}$
end while
(6) **if** $m \geq m_{\max}$ **then**
for $i = 2, .., m_{\min}$
$v_i = V s_i$, $v_i^A = V^A s_i$
$M =$ leading $m_{\min}$ by $m_{\min}$ block of T
$v_1 = u$, $v_1^A = u^A$, $m = m_{\min}$
end if
(7) $\widetilde{Q} = [Q, u]$, Solve $t\,(\perp \widetilde{Q})$ (approximately) from:
$(I - \widetilde{Q}\widetilde{Q}^*)(A - \theta I)(I - \widetilde{Q}\widetilde{Q}^*) t = -\widetilde{r}$

Figure 35: Jacobi-Davidson for $k_{\max}$ exterior eigenvalues (JDQR)

To apply this algorithm we need to specify a starting vector v_0, a tolerance ϵ, a target value τ, and a number $k_{\max}$ that specifies how many eigenpairs near τ should be computed. The value of $m_{\max}$ denotes the maximum dimension of the search subspace. If it is exceeded then a restart takes place with a subspace of specified dimension $m_{\min}$.

On completion, $k_{\max}$ eigenvalues close to τ are delivered, and the corresponding partial Schur form $AQ = QR$, where Q is n by $k_{\max}$ unitary and R is $k_{\max}$ by $k_{\max}$ upper triangular. The eigenvalues are on the diagonal of R. The computed Schur form satisfies $\|Aq_j - QRe_j\|_2 \leq j\epsilon$, where q_j is the jth column of Q.

We will now discuss some components of the algorithm.

(1) The new vector t is made orthogonal with respect to the current search subspace by means of modified Gram-Schmidt. This can be replaced, for improved numerical stability, by the template, for iterated modified Gram-Schmidt, given in Alg. 34.

(2) We compute the last column and row of the dense matrix $M := V^*AV = V^*V^A$ (of order m); $V^A := AV$. The matrix V denotes the n by m matrix with columns v_j, V^A likewise.

(3) The Schur reduction for the m by m matrix M can be solved by a standard solver for dense eigenproblems. We have chosen to compute the standard Ritz values, which makes the algorithm suitable for computing $k_{\max}$ exterior eigenvalues of A close to a specified τ. If eigenvalues in the interior part of the spectrum have to be computed, then the usage of harmonic Ritz values is advocated, see Section 49.

    In each step, the Schur form has to be sorted such that $T_{1,1}$ is closest to τ. Only when $m \geq m_{\max}$, the sorting of the Schur form has to be be carried out so that all of the $m_{\max}$ leading diagonal elements of T are the closest ones to τ. For ease of presentation we sorted all diagonal elements here.

    For a template of an algorithm for the sorting of a Schur form, see Van Dooren [1981a], Van Dooren [1982], Bai and Demmel [1993], and Fokkema [1996, Chapter 6B].

    S is the m by m matrix with columns s_j.

(4) We accept a Schur vector approximation when the norm of the residual (for the normalized Schur vector approximation) is below ϵ. This means that we accept inaccuracies in the order of ϵ in the computed eigenvalues, and inaccuracies (in angle) in the Schur vectors of $\mathcal{O}(\epsilon)$ (provided that the concerned eigenvalue is simple and well separated from the others).

    If the matrix is real, then all eigenpairs are either real or appear in complex conjugate pairs. If a complex eigenpair has been detected, then its conjugate is known and can be deflated as well. This makes the algorithm more efficient.

(5) After acceptance of a Ritz pair, we continue the search for a next pair, with the remaining Ritz vectors as a basis for the initial search space.

(6) We restart as soon as the dimension of the search space for the current Schur vector exceeds m_{max}. The process is restarted with the subspace spanned by the m_{min} Ritz vectors corresponding to the Ritz values closest to the target value τ.

(7) We have collected the locked (computed) Schur vectors in Q, and the matrix $\widetilde{Q}$ is Q expanded with the current Schur vector approximation u. This is done in order to obtain a more compact formulation; the correction equation is equivalent to the one in (46.1). The new correction t has to be orthogonal to the columns of Q as well as to u.

    Of course, the correction equation can be solved by any suitable process, for instance a preconditioned Krylov subspace method that is designed to solve unsymmetric systems. Make sure that the starting vector t_0 for an iterative solver satisfies the orthogonality constraints $\widetilde{Q}^* t_0 = 0$.

    It is not necessary to solve the correction equation very accurately. A strategy, often used for inexact Newton methods (DEMBO, EISENSTAT and STEIHAUG [1982]), here also works well: increase the accuracy with the Jacobi-Davidson iteration step, and, for instance, solve the correction equation with a residual reduction of $2^{-\ell}$ in the ℓ-th Jacobi-Davidson iteration (ℓ is reset to 0 when a Schur vector is detected). This strategy is inspired by the observation that the Jacobi-Davidson processes can be viewed as (inexact) Newton processes.

    In particular, in the first few initial steps, the approximate eigenvalue θ may be very inaccurate and then it does not make sense to solve the correction equation accurately. In this stage it can be more effective to temporarily replace θ by τ or to take $t = -r$ for the expansion of the search subspace (MORGAN and SCOTT [1986], FOKKEMA, SLEIJPEN and VAN DER VORST [1998]).

For a full theoretical background of this method, as well as details on the deflation technique with Schur vectors, see FOKKEMA, SLEIJPEN and VAN DER VORST [1998].

49 Harmonic Ritz Values

For restart purposes, specially if one is heading for interior eigenvalues, the harmonic Ritz vectors have been advocated (for symmetric matrices this has been advocated in MORGAN [1991]. In this context, we refer to our discussion on the harmonic Ritz values for the two-sided Lanczos method, see Section 36.

The concept of harmonic Ritz values (PAIGE, PARLETT and VAN DER VORST [1995]) is easily generalized for more general situations (SLEIJPEN and VAN DER VORST [1996a]). As we have seen, the Jacobi-Davidson methods generate basis vectors $v_1, \ldots, v_m$, for a subspace $\mathcal{V}_m$. For the projection of A onto this subspace we have to compute the vectors $w_j := Av_j$. The inverses of the Ritz values of A^{-1}, with respect to the subspace spanned by the w_j, are called the *harmonic Ritz values.* The harmonic Ritz values can be computed without inverting A, since a harmonic Ritz pair $(\widetilde{\theta}_j^{(m)}, \widetilde{u}_j^{(m)})$ satisfies

$$A\widetilde{u}_j^{(m)} - \widetilde{\theta}_j^{(m)}\widetilde{u}_j^{(m)} \perp \mathcal{W}_m := \mathrm{span}(w_1, \ldots, w_m), \qquad (49.1)$$

for $\widetilde{u}_j^{(m)} \in \mathcal{V}_m$ and $\widetilde{u}_j^{(m)} \neq 0$. This implies that the harmonic Ritz values are the eigenvalues of the pencil $(W_m^* AV_m, W_m^* V_m)$:

$$W_m^* AV_m s_j^{(m)} - \widetilde{\theta}_j^{(m)} W_m^* V_m s_j^{(m)} = 0\,.$$

The exterior standard Ritz values usually converge to exterior eigenvalues of A. Likewise, the interior harmonic Ritz values for the shifted matrix $A - \tau I$ usually converge towards eigenvalues $\lambda \neq \tau$ closest to the shift τ. Fortunately, the search subspaces $\mathcal{V}_m$ for the shifted matrix and the unshifted matrix coincide, which facilitates the computation of harmonic Ritz pairs. For reasons of stability we construct V_m and W_m so that they are both orthonormal: W_m is such that $(A - \tau I)V_m = W_m M_m^A$ where M_m^A is m by m upper triangular.

As has been shown in SLEIJPEN, VAN DER VORST and MEIJERINK [1998, Section 4.2], the harmonic Ritz vectors may carry more useful information than the harmonic Ritz values (cf. Section sec:alter). In particular, when an approximate eigenvector makes a small angle with an eigenvector and when the target is very close to the corresponding eigenvalue, then the harmonic Ritz value may be poor. This is because the harmonic eigenpair $\widetilde{\theta}_j^{(m)}, \widetilde{u}_j^{(m)}$ leads to a residual

$$r = A\widetilde{u}_j^{(m)} - \widetilde{\theta}_j^{(m)}\widetilde{u}_j^{(m)},$$

for which we have, cf. (49.1):

$$r \perp (A - \tau I)\widetilde{u}_j^{(m)}.$$

Consequently, if $\widetilde{u}_j^{(m)}$ and τ represent a very good approximation for an eigenpair of A, then $\widetilde{\theta}_j^{(m)}$ has to be a poor approximation. For a geometrical interpretation, see Section 25. Therefore, it has been suggested to base restart strategies on the Rayleigh quotients associated with the harmonic Ritz vectors, instaed of harmonic Ritz values.

Furthermore, we advocate to compute a (partial) Schur decomposition rather than a partial eigendecomposition. That is, we wish to compute vectors q_1, ..., q_k, such that $AQ_k = Q_k R_k$, with $Q_k^* Q_k = I_k$ and R_k is a k by k upper triangular matrix. The diagonal elements of R_k represent eigenvalues of A, and the corresponding eigenvectors of A can be computed from Q_k and R_k.

An algorithm for Jacobi-Davidson, with partial reduction to Schur form, and based on harmonic Ritz values and vectors, is given in BAI *et al* [2000]. That algorithm includes restart and deflation techniques. It can be used for the computation of a number of eigenvalues close to τ.

For an algorithm of the sorting of a generalized Schur form, see VAN DOOREN [1981a], VAN DOOREN [1982], BAI and DEMMEL [1993], and FOKKEMA [1996, Chapter 6B].

The stopping criterion is to accept a Schur vector approximation as soon as the norm of the residual (for the normalized Schur vector approximation) is below ϵ. This means that we accept inaccuracies in the order of ϵ in the computed eigenvalues, and inaccuracies (in angle) in the eigenvectors of $\mathcal{O}(\epsilon)$ (provided that the concerned eigenvalue is simple and well separated from the others and the left and right eigenvector have a small angle).

The deflation with computed Schur vectors is represented by the factors with Q. The matrix Q has the converged Schur vectors as its columns. If a left preconditioner K is available for the operator $A - \theta I$, then with a Krylov solver similar reductions are realizable as in the situation for exterior eigenvalues.

Chapter X

Generalized Eigenproblems

50 Preliminary remarks

In this chapter we will pay more attention to the possibilities for the iterative computation of (*right*) eigenpairs of the generalized non-Hermitian eigenvalue problem

$$Ax = \lambda Bx. \tag{50.1}$$

where A and B are general n by n matrices.

The perturbation theory for the generalized eigenvalue problem is considerably more complicated than the corresponding theory of the ordinary eigenvalue problem. We have briefly touched this in Chapter 12; a comprehensive treatment is given in Stewart and Sun [1990].

In Section 12, the QZ algorithm for the generalized eigenvalue problem has been discussed as a 'direct' approach. This is currently the most powerful method for dense problems, but it is only suitable for small to moderate size problems because the QZ method requires $\mathcal{O}(n^3)$ flops and $\mathcal{O}(n^2)$ memory locations.

A common approach for a large scale generalized eigenvalue problem is to reduce the problem (50.1) to a standard eigenvalue problem, and then apply an appropriate iterative method, for instance, Arnoldi's method, two-sided Lanczos, or the Jacobi-Davidson method. This reduction to standard form requires, for each iteration, the solution of a linear system with A, or B, or a combination of A and B. This will be discussed in more detail in Section 51.

The Rational Krylov Sequence (RKS) method, to be discussed in Section 52, is a generalization of the shift-and-invert Arnoldi method. It follows the basic ideas of the QZ method. In this approach it is permitted to work with different shifts for each iteration, without destroying the structure of the Krylov subspaces, which permits to adapt these shifts to the approximate eigenvalues that are delivered in each phase of the iteration process. In this method one has to solve a linear system with a matrix $A - \mu_j B$.

The Jacobi-Davidson method for generalized eigenproblems, presented in Section 53, avoids the transformation of $Ax = \lambda Bx$ to a standard eigenproblem. For this method also a linear system has to be solved for each iteration, but this may be done only approximately, which offers possibilities to improve overal efficiency.

51 Reduction to Standard Form

An obvious approach for the numerical solution of the generalized eigenvalue problem (50.1) is to reduce the problem first to an equivalent standard eigenvalue problem, and then apply an appropriate iterative method for the standard eigenproblem. Here we will discuss three approaches for the reduction to a standard eigenproblem.

1. **Inverse** B: If the matrix B is nonsingular and well conditioned, then we can do operations with the inverse of B. The problem (50.1) is equivalent to

$$(B^{-1}A)x = \lambda x. \tag{51.1}$$

   For an iterative method, such as we have described in previous chapters, for the reduced standard eigenvalue problem (51.1) it is not necessary to evaluate the product $B^{-1}A$. This is important, since A and B are in most applications sparse, and forming the matrix $B^{-1}A$ (most likely a dense matrix) would have been very very unattractive.

   One only needs to evaluate matrix-vector products, like

$$p = (B^{-1}A)q$$

   for a given vector q.

   The computation of a vector $v = B^{-1}y$ is done via a (sparse) LU factorization of B and then solving v from $Bv = y$.

   The error introduced by this reduction to standard form can be proportional to $\|A\|_2\|B^{-1}\|_2$. If B is ill-conditioned, then the approach is potentially suspect. In that situation, one may consider the usage of the shift-and-invert transformation for the reduction (see (51.3)), the Cayley Transform (see (51.5)), or the usage of the Jacobi-Davidson method.

   It may be tempting to consider iterative solution methods for the approximate solution of systems like $Bz = u$, instead of the possibly expensive LU factorization. However, if one does so then one should realize that the approximate solution satisfies some nearby system, say with a matrix $\widetilde{B}$. This nearby matrix may differ significantly from iteration to iteration in forming the (Krylov) subspace for the eigenvalue method. This will most likely destroy the structure of the Krylov subspace. Therefore, an iterative solution technique is only recommended if it leads efficiently to high accuracy, comparable with a stable direct solution method. Likewise, if the direct solution method has made a compromise between stability and sparsity (to the advantage of the latter), for instance, by working with threshold pivoting (DUFF, ERISMAN and REID [1986]), it is advisable to use iterative refinement in order to get more accurate solutions (GOLUB and VAN LOAN [1996]).

2. **Symmetrized inverse** B: In some applications, B is Hermitian positive definite and well-conditioned. In this case, it is recommended to compute first a sparse Cholesky decomposition

$$B = LL^*,$$

with L a lower triangular matrix. The equivalent standard eigenvalue problem is

$$(L^{-1}AL^{*-1})y = \lambda y, \tag{51.2}$$

where $y = L^*x$. As in the 'Invert B' approach, matrices like $L^{-1}AL^{*-1}$ should never be evaluated explicitly, since they will in general not be sparse. For the application of an iterative method for the standard eigenvalue problem, we only need to provide the efficient evaluation of matrix-vector products, like

$$p = (L^{-1}AL^{*-1})q.$$

where q is a given vector.

Since L is a triangular matrix, the solutions of linear systems with L or L^* can be obtained by forward and backward substitutions. Sparsity can be exploited in a straightforward manner in all these steps.

3. **Shift-and-Invert**: The reductions to standard form (51.1) or (51.2) cannot be used when B is singular or ill-conditioned. An attractive and popular technique is to apply first the shift to the original problem and then carry out the reduction. This is the so-called *shift and invert spectral transformation.* Specifically, let σ be a user-selected shift such

that the matrix $A - \sigma B$ is nonsingular, then the original problem (50.1) can be transformed to

$$Cx = \mu x, \tag{51.3}$$

where

$$C = (A - \sigma B)^{-1} B \quad \text{and} \quad \mu = \frac{1}{\lambda - \sigma},$$

We see that the eigenvalues λ of the problem (50.1) closest to the shift σ are mapped as the exterior eigenvalues of the reduced standard eigenvalue problem (51.3), that is to the eigenvalues of largest magnitude, and these are the eigenvalues that are first well approximated by most iterative methods.

In practice, an effective shift selection depends on the user's preferences and on knowledge of the underlying generalized eigenproblem. A good shift not only amplifies the desired eigenvalues, but it also leads to a well-conditioned matrix $A - \sigma B$. This makes the task of selecting good shifts often a challenging one.

For the application of an iterative method for the reduced standard eigenvalue problem (51.3), one needs to evaluate matrix-vector products

$$p = Cq = \left((A - \sigma B)^{-1} B\right) q,$$

for a given vector q.

For an efficient evaluation, we make an LU factorization

$$A - \sigma B = \mathcal{L}\mathcal{U}, \tag{51.4}$$

where $\mathcal{L}$ and $\mathcal{U}$ are square matrices. Since $A - \sigma B$ is assumed to be nonsingular, the factors $\mathcal{L}$ and $\mathcal{U}$ are also nonsingular. The factorization should be chosen so that the corresponding linear systems of equations with $\mathcal{L}$, $\mathcal{L}^*$ and/or $\mathcal{U}$, $\mathcal{U}^*$ can be solved efficiently, and typically, sparse LU factorizations are used. Of course, one can also select $\mathcal{L} = A - \sigma B$ and $\mathcal{U} = I$ if this leads to convenient linear systems.

With the above factorization, the matrix-vector product $p = Cq$ can be evaluated as follows:

(a) form $v = Bq$
(b) solve $\mathcal{L}w = v$ for w
(c) solve $\mathcal{U}p = w$ for p.

4. **Cayley Transform**: With Shift-and-Invert it is possible to emphasize the converge towards eigenvalues close to the shift σ. The Cayley Transform goes one step further. It is defined by the operator

$$(A - \sigma B)^{-1}(A - \tau B).$$

To be more specific, let σ and τ be a user-selected shifts such that the matrix $A - \sigma B$ is nonsingular, then the original problem (50.1) can be transformed to

$$Cx = \mu x, \tag{51.5}$$

where

$$C = (A - \sigma B)^{-1}(A - \tau B) \quad \text{and} \quad \mu = \frac{\lambda - \tau}{\lambda - \sigma}.$$

The efficient implementation of the Cayley Transform can be done along similar lines as the Shift-and-Invert transformation. With the Cayley Transform we can emphasize the convergence towards eigenvalues close to σ while suppressing the influence of eigenvalues close to τ. It is a special form of the more general polynomial preconditioners by which we can accelerate the convergence towards wanted parts of the spectrum. For more details and references, see SAAD [1992].

The shift and invert spectral transformation technique is a powerful tool in the treatment of the generalized eigenvalue problem (50.1). The major problem, which often becomes a bottleneck, is to find a convenient factorization (51.4) of $A - \sigma B$ so that the associated linear systems of equations can be solved efficiently. If accurate solution of the linear systems with $A - \sigma B$ becomes to expensive, then one may consider the usage of the Jacobi-Davidson method, or the application of inexact Cayley transforms (LEHOUCQ and MEERBERGEN [1998], MEERBERGEN [1996], BAI *et al* [2000]).

52 Rational Krylov Methods

The Arnoldi iteration is often carried out with the shift-and-invert approach. For instance, when solving the generalized eigenproblem $Ax = \lambda Bx$, the method is applied to the operator $(A - \sigma B)^{-1}B$. It requires possibly expensive operations with an inverted operator but the advantage is much faster convergence. MEERBERGEN [1996] (see also MEERBERGEN and MORGAN [2000]) considers the use of inexact Cayley transforms, realized by a few steps of an iterative method, for Arnoldi's method. This technique has a close relation with polynomial preconditioning. RUHE [1994b] considers a slightly more general shift-and-invert transform, for the construction of a so-called *Rational Krylov Sequence (RKS)*:

$$(\delta_j A - \gamma_j B)^{-1}(\sigma_j A - \rho_j B),$$

in which the coefficients may be different for each iteration step j.

It has been shown that by generating a subspace with this operator, the given problem can be reduced to a small projected generalized system

$$(\zeta K_{j,j} - \eta L_{j,j})s = 0,$$

where $K_{j,j}$ and $L_{j,j}$ are upper Hessenberg matrices of dimension j. This small system may be solved by the QZ algorithm in order to obtain approximate values for an eigenpair. The parameters in RKS can be chosen to obtain faster convergence to interior eigenvalues. For a comparison of RKS and Arnoldi, see Ruhe [1994a] and Ruhe [1994b].

The big advantage of Ruhe's RKS approach over working with Arnoldi with a fixed transform, is the possibility to select appropriate different shifts in each iteration step. The method has a close resemblance to the initial Hessenberg step in the QZ method (see Chapter 12). We will explain the RKS method for the simplified situation where we work with the shift-and-invert transform.

With the shift-and-invert transform we can apply Arnoldi's method, which generates a Krylov subspace for the matrix $(A - \sigma B)^{-1}B$ with starting vector v_1. More in particular, in the mth step of this algorithm, we have arrived at a basis v_1, v_2, ..., v_m of the Krylov subspace. As usual, we collect these basis vectors in a matrix V_m. The new vector v_{m+1} is formed by putting $(A - \sigma B)^{-1}Bv_m$ orthogonal to the previous basis vectors. The Gram-Schmidt orthogonalization (see Figure 5) leads to the mth column of the Arnoldi upper Hessenberg matrix:

$$h_{\cdot,m} = (h_{1,m}, h_{2,m}, \ldots, h_{m+1,m})^T.$$

We can write this in matrix vector notation as

$$V_{m+1}h_{\cdot,m} = (A - \sigma B)^{-1}Bv_m. \tag{52.1}$$

This expression is now multiplied with $A - \sigma B$. The parts with A are kept at the left side, those with B are moved to the right side. In order to simplify the resulting expressions, we define $t_{\cdot,m} \in \mathbb{C}^{m+1}$ as the mth canonical basis vector, expanded with one zero

$$t_{\cdot,m} = (0, 0, \ldots, 1, 0)^T.$$

Note that $v_m = V_{m+1}t_{\cdot,m}$.

By doing so, the following expression is obtained

$$AV_{m+1}h_{\cdot,m} = BV_{m+1}(\sigma h_{\cdot,m} + t_{\cdot,m}).$$

For all previous vectors together, this expression becomes

$$AV_{m+1}H_{m+1,m} = BV_{m+1}K_{m+1,m}.$$

The nice consequence is that $K_{m+1,m} = \sigma H_{m+1,m} + I_{m+1,m}$ is also upper Hessenberg. Going back once again over this derivation, we see that we could have made it more general with little effort. In particular, we could have selected a different σ_j for step j, and we could have taken $t_{\cdot,m} \in \mathbb{C}^{m+1}$ as a general vector with

$$t_{m,m} \neq 0 \quad \text{and} \quad t_{m+1,m} = 0.$$

With this vector $t_{\cdot,m}$ we continue an Arnoldi step, using (52.1), with a combination of all previous basis vectors, instead of v_m. This has advantages for the generalized eigenproblem, as we will see later. This leads to

$$AV_{m+1}H_{m+1,m} = BV_{m+1}(H_{m+1,m}\Sigma_m + T_{m+1,m}), \tag{52.2}$$

where $\Sigma_m = \text{diag}(\sigma_i)$ is a diagonal matrix, and $T_{m+1,m}$ is upper triangular, so that

$$K_{m+1,m} = H_{m+1,m}\Sigma_m + T_{m+1,m}$$

is still upper Hessenberg.

With a proper choice of the vectors $t_{\cdot,m}$ we can influence the matrix $K_{m+1,m}$ in order to make the reduced generalized eigenproblem in a better conditioned one.

We are now in the position to solve a reduced generalized eigenproblem, for vectors

$$p \in \text{span}\{v_1, v_2, \ldots, v_{m+1}\}.$$

First note that A times a vector in this space is mapped onto B times a vector in that space, on account of (52.2), and vice versa. In other words, both A and B map this space onto itself. Therefore, we can obtain meaningful approximations by solving the reduced generalized eigenproblem

$$(K_{m,m} - \theta^{(m)}H_{m,m})s^{(m)} = 0, \tag{52.3}$$

with the QZ method (see Chapter 12).

For a computed solution $\theta_j^{(m)}, s_j^{(m)} \in \mathbb{C}^m$, we form the Ritz vector

$$u_j^{(m)} = V_{m+1} H_{m+1,m} s_j^{(m)}. \tag{52.4}$$

For the resulting Ritz pair we can compute, for relatively low costs, the residual vector $r_j^{(m)} = (A - \theta_j^{(m)} B) u_j^{(m)}$:

$$\begin{aligned} r_j^{(m)} &= (A - \theta_j^{(m)} B) V_{m+1} H_{m+1,m} s_j^{(m)} \\ &= B V_{m+1} (K_{m+1,m} - \theta_j^{(m)} H_{m+1,m}) s_j^{(m)} \\ &= B v_{m+1} (k_{m+1,m} - \theta_j^{(m)} h_{m+1,m}) s_{m,j}^{(m)} \\ &= B v_{m+1} (\sigma_m - \theta_j^{(m)}) h_{m+1,m} s_{m,j}^{(m)} \end{aligned} \tag{52.5}$$

where $s_{m,j}^{(m)}$ denotes the last coordinate of the vector $s_j^{(m)}$.

Note that the expression at the right-hand side of (52.5) can be evaluated with information that is obtained from the (relatively) small reduced system and the vector Bv_{m+1}. This vector has to be computed in the mth step of the Arnoldi algorithm, and should be saved for this residual vector computation.

The derivation of the more general RKS algorithms follows the lines of the above exposition. For further details and practical choices of the included parameters, we refer to RUHE [1994a] and RUHE [1994b].

53 The Jacobi-Davidson QZ method

Basic Theory Similar as for the generalized Hermitian eigenproblem, we want to avoid transformation of $Ax = \lambda Bx$ to a standard eigenproblem. This would require the solution of some linear system, involving B and/or A, per iteration step. Furthermore, for stability reasons we want to work with orthonormal transformations and for this reason our goal is to compute Schur vectors for the pencil $A - \lambda B$, rather than eigenvectors. This leads to a generalization of the Jacobi-Davidson algorithm, in which we compute a partial Schur form for the pencil. This algorithm can be interpreted as a subspace iteration variant of the QZ-algorithm. A consequence of this approach is that we have to work with search and test spaces that are different.

With $\lambda = \alpha/\beta$, the generalized eigenproblem $Ax = \lambda Bx$ is equivalent to the eigenproblem

$$(\beta A - \alpha B)x = 0, \tag{53.1}$$

and we denote a generalized eigenvalue of the matrix pair (A, B) as a pair (α, β). This approach is preferred, because underflow or overflow for $\lambda = \alpha/\beta$ in finite precision arithmetic may occur when α and/or β are zero or close to zero, in which case the pair is still well-defined (cf Moler and Stewart [1973], Stewart and Sun [1990, Chapter VI], Ruhe [1994a].

A *partial generalized Schur form* of dimension k for a matrix pair (A, B) is the decomposition

$$AZ_k = Q_k R_k^A, \quad BZ_k = Q_k R_k^B, \tag{53.2}$$

where Z_k and Q_k are unitary n by k matrices, and R_k^A and R_k^B are upper triangular k by k matrices. A column z_i of Z_k is referred to as a *generalized Schur vector*, and we refer to a pair $((\alpha_i, \beta_i), z_i)$, with $(\alpha_i, \beta_i) = (R_k^A(i,i), R_k^B(i,i))$ as a *generalized Schur pair*.
It follows that if $((\alpha, \beta), y)$ is a generalized eigenpair of (R_k^A, R_k^B) then

$$((\alpha, \beta), Z_k y)$$

is a generalized eigenpair of (A, B).

We will now describe the Jacobi-Davidson method for the generalized eigenproblem (53.1). From the relations (53.2) we conclude that

$$\beta_i A z_i - \alpha_i B z_i \perp q_i,$$

which suggests that we should follow a Petrov-Galerkin condition for the construction of reduced systems. In each step the approximate eigenvector u is selected from a *search subspace* $\text{span}(V)$. We require that the residual $\eta A u - \zeta B u$ is orthogonal to some other well-chosen *test subspace* $\text{span}(W)$:

$$\eta\, Au - \zeta\, Bu \perp \text{span}(W). \tag{53.3}$$

The pair (η, ζ) can be interpreted as a generalized Petrov value; if $\zeta \neq 0$, and $\beta \neq 0$, then η/ζ is an approximation for α/β for some generalized eigenpair (α, β). Both subspaces are of the same dimension, say m. Equation (53.3) leads to the *projected eigenproblem*

$$(\eta\, W^*AV - \zeta\, W^*BV)\, s = 0. \tag{53.4}$$

The pencil $\eta\, W^*AV - \zeta\, W^*BV$ can be reduced by the QZ algorithm (Moler and Stewart [1973]) to generalized Schur form (note that this is a m-dimensional problem). This leads to orthogonal m by m matrices S^R and S^L, and upper triangular m by m matrices T^A and T^B, such that

$$(S^L)^*(W^*AV)S^R = T^A \quad \text{and} \quad (S^L)^*(W^*BV)S^R = T^B. \tag{53.5}$$

The decomposition can be reordered such that the first column of S^R and the $(1,1)$-entries of T^A and T^B represent the wanted Petrov solution (FOKKEMA, SLEIJPEN and VAN DER VORST [1998]).
With $s := s_1^R := S^R e_1$ and $\zeta := T^A_{1,1}$, $\eta := T^B_{1,1}$, the *Petrov vector* is defined as $u := Vs$, for the associated *generalized Petrov value* (ζ, η). The vector u is an approximation to the Schur vector q_1, and (η, ζ) is an approximation to the generalized eigenvalue (α_1, β_1).

In the actual algorithm, one constructs orthogonal matrices V and W, so that also $\|u\|_2 = 1$.
With the decomposition in (53.5), we construct an approximate partial generalized Schur form (cf. (53.2)): VS^R approximates a Z_k, and WS^L approximates the associated Q_k. Since $\mathrm{span}(Q_k) = \mathrm{span}(AZ_k) = \mathrm{span}(BZ_k)$ (cf. (53.2)), this suggests to choose W such that $\mathrm{span}(W)$ coincides with $\mathrm{span}(\nu_0 AV + \mu_0 BV)$, for certain scalars ν_0 and μ_0. With the weights ν_0 and μ_0 we can influence the convergence of the Petrov values. If we want eigenpair approximations for eigenvalues λ close to a target τ, then the choice

$$\nu_0 = 1/\sqrt{1+|\tau|^2} \quad , \quad \mu_0 = -\tau\nu_0$$

is very effective (FOKKEMA, SLEIJPEN and VAN DER VORST [1998]), especially if we want to compute eigenvalues in the interior of the spectrum of $A - \lambda B$. We will call the Petrov approximations for this choice the *harmonic Petrov* eigenpairs.
The Jacobi-Davidson correction equation for the component $t \perp u$, for the pencil $\eta A - \zeta B$ becomes:

$$\left(I - \frac{pp^*}{p^*p}\right)(\eta A - \zeta B)(I - uu^*)\,t = -r, \tag{53.6}$$

with $r := \eta Au - \zeta Bu$, and $p := \nu_0 Au + \mu_0 Bu$. It can be shown that if (53.6) is solved exactly, the convergence to the generalized eigenvalue will be quadratic, see SLEIJPEN, BOOTEN *et al* [1996, Theorem 3.2].
Usually, this correction equation is solved only approximately, for instance, with a (preconditioned) Krylov subspace method, such as GMRES. A preconditioner $K \approx (\eta A - \zeta B)^{-1}$ can be included in a similar way as for the standard eigenproblem (for further details on this, see FOKKEMA, SLEIJPEN and VAN DER VORST [1998]. The obtained vector t is used for the orthogonal expansion v of V and $\nu_0 Av + \mu_0 Bv$ is used for the expansion of W. For both spaces we work with orthonormal bases. Therefore, the new columns are orthonormalized with respect to the current basis by modified Gram-Schmidt, as in Algorithm 5, or if necessary with refined classical Gram-Schmidt (see Fig. 34).

It can be shown that, with the above choices for p and W,

$$p = W s_1^L = W S^L e_1. \tag{53.7}$$

The relation between the partial generalized Schur form for the given large problem and the complete generalized Schur form for the reduced problem (53.4), via right vectors ($u = V s_1^R$), is similar to the relation via left vectors ($p = W s_1^L$). This can also be exploited for restarts.

Deflation and Restart Similar to the situation for the standard eigenproblem, see Chapter 46, the Jacobi-Davidson process can be carried out with restarts in order to restrict the dimension of the search subspace. The process can also be combined with deflation in order to find a number of different generalized Schur pairs.

Deflation The partial generalized Schur form can be obtained in a number of successive steps. Suppose that we have the partial generalized Schur form $AZ_{k-1} = Q_{k-1} R^A_{k-1}$ and $BZ_{k-1} = Q_{k-1} R^B_{k-1}$. We want to expand this partial generalized Schur form with the new right Schur vector z and the left Schur vector q to

$$\begin{aligned} A\,[Z_{k-1}, z] &= [Q_{k-1}, q] \begin{pmatrix} R^A_{k-1} & a \\ 0 & \alpha \end{pmatrix} \\ \text{and} \quad B\,[Z_{k-1}, z] &= [Q_{k-1}, q] \begin{pmatrix} R^B_{k-1} & b \\ 0 & \beta \end{pmatrix}. \end{aligned} \tag{53.8}$$

The new generalized Schur pair $((\alpha, \beta), z)$ satisfies

$$Z^*_{k-1} z = 0 \quad \text{and} \quad (\beta A - \alpha B) z - Q_{k-1}(\beta a - \alpha b) = 0, \tag{53.9}$$

or, since $\beta a - \alpha b = Q^*_{k-1}(\beta A - \alpha B) z$,

$$Z^*_{k-1} z = 0 \quad \text{and} \quad \left(I - Q_{k-1} Q^*_{k-1}\right)(\beta A - \alpha B)\left(I - Z_{k-1} Z^*_{k-1}\right) z = 0. \tag{53.10}$$

The vectors a and b can be computed from

$$a = Q^*_{k-1} A q \quad \text{and} \quad b = Q^*_{k-1} B z. \tag{53.11}$$

Hence, the generalized Schur pair $((\alpha, \beta), z)$ is an eigenpair of the deflated matrix pair

$$\begin{aligned} \big(\left(I - Q_{k-1} Q^*_{k-1}\right) A \left(I - Z_{k-1} Z^*_{k-1}\right), \\ \left(I - Q_{k-1} Q^*_{k-1}\right) B \left(I - Q_{k-1} Q^*_{k-1}\right)\big) \end{aligned} \tag{53.12}$$

This eigenproblem can be solved again with the Jacobi-Davidson process that we have outlined in Section 53. In that process we construct vectors v_i that are orthogonal to Z_{k-1} and vectors w_i that are orthogonal to Q_{k-1}. This simplifies the computation of the interaction matrices M^A and M^B, associated with the deflated operators:

$$\left\{\begin{array}{l} M^A := W^*\left(I - Q_{k-1}Q_{k-1}^*\right) A \left(I - Z_{k-1}Z_{k-1}^*\right) V = W^*AV \\ M^B := W^*\left(I - Q_{k-1}Q_{k-1}^*\right) B \left(I - Z_{k-1}Z_{k-1}^*\right) V = W^*BV \end{array}\right. . \tag{53.13}$$

Restart Suppose that the generalized Schur form (53.5) is ordered with respect to τ such that

$$|T^A_{1,1}/T^B_{1,1} - \tau| \le |T^A_{2,2}/T^B_{2,2} - \tau| \le \ldots \le |T^A_{m,m}/T^B_{m,m} - \tau|, \tag{53.14}$$

where m is the dimension of $\mathrm{span}(V)$.

Then, for $i < m$, the space

$$\mathrm{span}(Vs^R_1, \ldots, Vs^R_i),$$

spanned by the first i columns of VS^R contains the i most promising Petrov vectors. The corresponding test subspace is given by $\mathrm{span}(Ws^L_1, \ldots, Ws^L_i)$. Therefore, in order to reduce the dimension of the subspaces ("implicit restart") to $m_{\min}$, $m_{\min} < m$, the last $m - m_{\min}$ columns of VS^R and WS^L can simply be discarded and the Jacobi-Davidson algorithm can be continued with

$$V = [Vs^R_1, \ldots, Vs^R_{m_{\min}}] \quad \text{and} \quad W = [Ws^L_1, \ldots, Ws^L_{m_{\min}}]. \tag{53.15}$$

A complete algorithm for the computation of a partial QZ factorization, with the Jacobi-Davidson method, based upon Harmonic Ritz values (better: Harmonic Petrov values) has been given in BAI *et al* [2000].

It is in many situations not necessary to solve the correction equation very accurately. A strategy, often used for inexact Newton methods (DEMBO, EISENSTAT and STEIHAUG [1983]), here also works well: increase the accuracy with the Jacobi-Davidson iteration step, for instance, solve the correction equation with a residual reduction of $2^{-\ell}$ in the ℓ-th Jacobi-Davidson iteration (ℓ is reset to 0 when a Schur vector is detected).

In particular, in the first few initial steps, the approximate eigenvalue θ may be very inaccurate and then it does not make sense to solve the correction equation accurately. In this stage it can be more effective to temporarily replace θ by τ or to take $t = -r$ for the expansion of the search

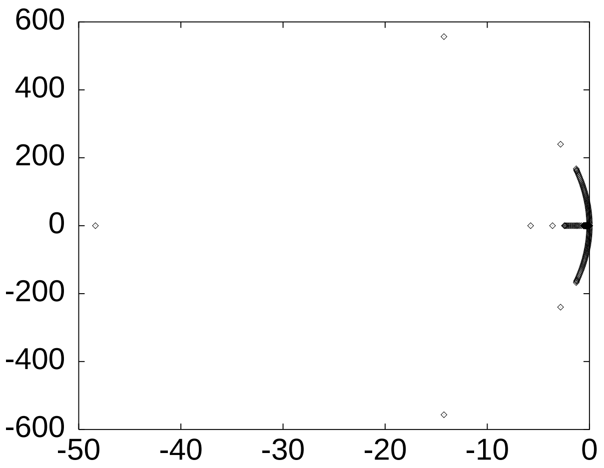

Figure 36: The complete spectrum of the matrix of order 416

subspace (Morgan and Scott [1986], Fokkema, Sleijpen and van der Vorst [1998]).

For a full theoretical background of this method, as well as for details on the deflation technique with Schur vectors and many numerical examples, see Fokkema, Sleijpen and van der Vorst [1998]. For a complete algorithm that incorporates the Jacobi-Davidson QZ algorithm, see Bai *et al* [2000].

54 An example of a generalized eigenproblem

Large generalized eigenproblems arise in many scientific problems. We consider as an example a problem from plasmaphysics. In this field one is interested in the instabilities of plasmas in a magnetic field with computational models. The results are applied for further understanding of the stability of Tokamak fusion reactors and of coronal loops, as well as of solar flares. A similar class of problems was also used to explain the usage of the Two-sided Lanczos method (Cullum, Kerner and Willoughby [1989]).

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

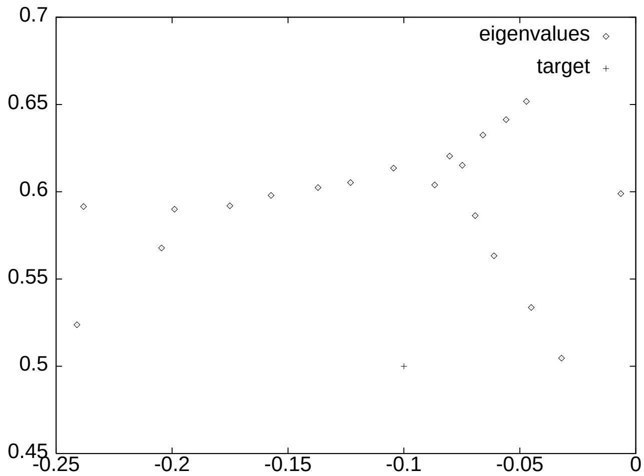

Figure 37: The Ritz values closest to the shift (+), as computed with the JDQZ algorithm

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

The last equation is considered as an initial condition on **B**. A familiar approache is to consider small perturbations of the unknowns with respect to some known equilibrium, and this leads to a linear system for 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 have used, partly finite differences, and partly Fourier series, this leads to block tridiagonal matrices, with typ-

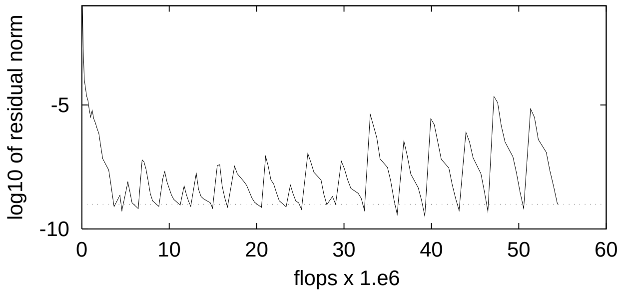

Figure 38: The convergence history of the JDQZ process

ically $100 - 2,000$ blocks, of size $80 \ - \ 320$ each. This amounts to matrices of orders in the range $8,000 \ - \ 640,000$. For specific details, we refer to Booten *et al* [1996].
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.

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 described in Cullum, Kerner and Willoughby [1989]) and by the Shift and Invert Implicitly Restarted Arnoldi method (Sorensen [1992]). The applicability of these methods is limited, due to the fact that inversion of many matrices is required, and this becomes too expensive for the very large problems that we are interested in. Accurate inversion of matrices is essential for these methods in order to obtain convergence for the Alfvén spectrum.

In view of the problems that one actually wants 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 Booten *et al* [1996]. In Fig. 36 we see the complete spectrum; Fig. 37 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. We have computed the 20 eigenvalue approximations, shown in Fig. 38, by the Jacobi-Davidson QZ algorithm. 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 $\tau = -.1 + .5i$.

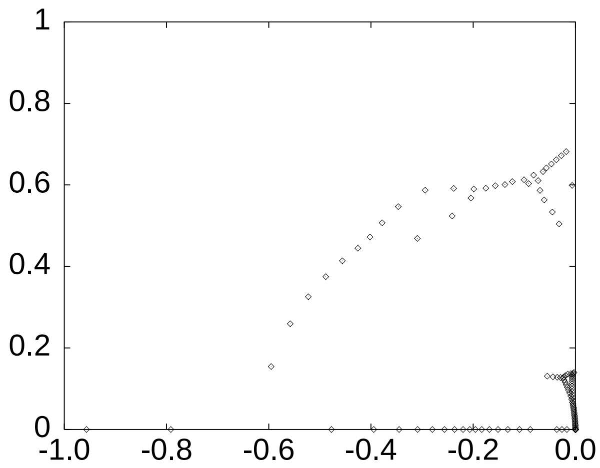

Figure 39: The Alfvén part of the spectrum

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 Chapter 53, 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 Fig. 39.
For more examples and a more thorough discussion, see FOKKEMA, SLEIJPEN and VAN DER VORST [1998].

NOOL and VAN DER PLOEG [1999] have used the Jacobi-Davidson approach for the (parallel) solution of Alfvén spectral information for matrices of order up to $n = 262144$, which is orders of magnitudes bigger than the matrices that could be handled by other methods.