Inverse iteration and deflation in general eigenvalue problems

M.R.Osborne and D.L. Harrar II
Centre for Mathematics and its Applications
School of Mathematical Sciences
Australian National University
Canberra, ACT 0200

March 6, 1997

Abstract

This paper develops Newton method based procedures for the general eigenvalue problem using techniques of inverse iteration and deflation. The approach has advantages when the problem matrix is narrow banded, when local information on the eigenvalue spectrum is required for large scale problems, when specific questions about particular eigenvalues are asked, and when high performance software for solving systems of linear equations is available.

1 Introduction

That inverse iteration provides an effective and stable method for eigenvector computation was shown in [7]. That effective choices of shift parameters giving second and third order convergence in algorithms based on inverse iteration and shift of origin can be based on Newton method based strategies is probably due to [5] where the application to nonlinear eigenvalue problems is noted also. A recent reference to aspects of the application of this class of methods to large scale problems is [1]. The key idea is that of replacing the eigenvalue problem by the problem of finding the zeros of a single nonlinear function. Let the eigenvalue problem be

$$M(\lambda)\mathbf{v} = 0, \tag{1}$$

where $M : C^n \to C^n$. Note that the eigenvalue problem is not required to be linear in the eigenvalue parameter λ. However, reasonable smoothness is required of the components of M. To construct the iteration function consider the system

$$M(\lambda)\mathbf{v} = \beta(\lambda)\mathbf{x}, \tag{2}$$

$$\mathbf{s}^T\mathbf{v} = \kappa. \tag{3}$$

This includes (1) in the following sense. If λ tends to a value which makes $M(\lambda)$ singular then the solution of (2) for almost all right hand sides will become unbounded. This means that the scaling condition (3) can be satisfied only if the scaling function $\beta(\lambda)$ tends to 0. It follows that this process generates a function $\beta(\lambda)$ which vanishes at the eigenvalues (values which make $M(\lambda)$ singular). To compute the function $\beta(\lambda)$ requires the solution of a linear system with matrix M, while successive iterations of Newton's method require shifts of the origin λ by the amount of the Newton corrections. Thus the similarity with strategies which use inverse iteration and shift of origin is very close, and many of these algorithms (possibly all methods with second order or better convergence rates which isolate a single eigenvalue at a time) can be interpreted as Newton strategies. There are considerable ranges of choice in developing these Newton strategies as $\mathbf{s}, \mathbf{x}$ are disposable. Also, they can be further selected at each iteration as this amounts to selecting a favourable member of a family of consistent iterative schemes. It is this freedom which permits these (composite) methods to achieve better than the characteristic second order rate for Newton's method. Well known examples include the various Rayleigh quotient iterations for symmetric eigenvalue problems, but the scope of these methods is much wider, extending to unsymmetric and nonlinear problems.

One advantage of these methods is that they are readily implementable on large scale computers given the availability of good software for solving systems of linear equations, and they make few requirements of the structure of the underlying problem - for example, the weighting matrix $\frac{dM}{d\lambda}$in the generalised eigenvalue problem is not required to be nonsingular. They are often suitable for answering specific questions about a small number of eigenvalues - for example, finding the zero of the real part of a particular eigenvalue as a function of an auxiliary parameter. Typically, they are not suitable for finding many eigenvalues of a full matrix. One problem is that M must be factored at each iteration. If M is full this is the expensive part of the calculation, and comparatively little use is made of each set of factors. As a consequence, in sequential computation, the method compares badly with the QR algorithm which uses inverse iteration implicitly in refining the

smallest eigenvalue while simultaneously improving estimates of other values. Also it can compare unfavourably with implementations of the rational Arnoldi method which can use a single matrix factorization to build up its Krylov subspace information. It suffers from the problems encountered by all methods which try to find multiple solutions to a general nonlinear equation. This is an important shortcoming, especially for linear problems, when significant local information on the spectrum is often required. In this case the linear structure can be exploited to deflate the problem, and one challenge is to avoid destroying band structure as a consequence of the deflation strategy.

The plan of the paper is as follows. In the next section the general Newton iteration is derived and its rate of convergence discussed for the generalised linear eigenvalue problem. It is shown that convergence rates of better than third order are achievable. The problem of computing a sequence of eigenvalues for the general unsymmetric generalised eigenvalue problem using complex arithmetic and a variant of Weilandt deflation is discussed in section 3. This approach has proved rather satisfactory, especially in the case of generalised eigenvalue problems for narrow band matrices when our vectorized narrow band solvers based on wrap-around partitioning [3] can be used. Here the cost of factorization does not differ by an order of magnitude from the cost of forward and back substitution. Also, our deflation strategy does not destroy band structure. This procedure is not so satisfactory for full matrices on parallel and vector computers because, in addition to the matrix factorization cost, knowledge of both right and left vectors is required, and this needs both row and column matrix access. An alternative based on the use of Schur-Weilandt deflation [6] has been tried with reasonable success. The inverse iteration Newton method has been applied also to the problem of extracting quadratic factors of the characteristic polynomial of a real matrix using real arithmetic. Computational results for this procedure have proved somewhat disappointing. The main results are summarised in an appendix.

2 The Newton iteration

Differentiation of (2), (3) with respect to λ gives

$$\begin{aligned} M\frac{d\mathbf{v}}{d\lambda} + \frac{dM}{d\lambda}\mathbf{v} &= \frac{d\beta}{d\lambda}\mathbf{x}, \\ \mathbf{s}^T\frac{d\mathbf{v}}{d\lambda} &= 0. \end{aligned}$$

Eliminating between these equations gives the Newton correction in the form

$$\frac{\beta}{d\beta/d\lambda} = \frac{\mathbf{s}^T\mathbf{v}}{\mathbf{s}^T M^{-1}\frac{dM}{d\lambda}\mathbf{v}}. \tag{4}$$

In [4] an indication of the range of special cases that can be obtained by specialising (4) is given. This reference also contains a discussion of convergence rates, including conditions under which third order convergence is obtained for quite general problems.

2.1 Application to the linear problem

We now specialise to the problem

$$(A - \lambda B)\,\mathbf{v} = 0. \tag{5}$$

It is assumed that $\mathcal{N}(A) \cap \mathcal{N}(B) = \emptyset$, so that the continuous spectrum is empty, and that $\dim(\mathcal{N}(B)) = n - p$. Let the eigen decomposition of (5) be

$$AV = BV\Lambda,\ S^TA = \Lambda S^TB,\ S^TBV = I_p$$

where $V, S : C^p \to C^n$, and $\Lambda : C^p \to C^p$, diagonal (so an implicit assumption of a nondefective problem is made). Also, let $Y : C^{n-p} \to C^n$ be a basis for $\mathcal{N}(B)$. Then any vector can be written

$$\mathbf{x} = V\mathbf{z}_1 + Y\mathbf{z}_2$$

where

$$\mathbf{z}_1 = S^TB\mathbf{x}.$$

The iteration function is defined by the linear system

$$\begin{aligned} (A - \lambda B)\,\mathbf{v} &= \beta(\lambda)B\mathbf{x}, & (6)\\ \mathbf{s}^TB\mathbf{v} &= \kappa, & (7) \end{aligned}$$

(note the placing of the weighting matrix B) giving

$$\begin{aligned} \mathbf{v} &= \beta(\lambda)V\,(\Lambda - \lambda I)^{-1}\,\mathbf{z}_1, & (8)\\ \beta(\lambda) &= \frac{\kappa}{\mathbf{s}^TBV\,(\Lambda - \lambda I)^{-1}\,\mathbf{z}_1}. & (9) \end{aligned}$$

An important intermediate vector $\mathbf{w}$ is given by

$$\begin{aligned} (A - \lambda B)\,\mathbf{w} &= B\mathbf{v},\\ \mathbf{w} &= \beta(\lambda)V\,(\Lambda - \lambda I)^{-2}\,\mathbf{z}_1. \end{aligned}$$

Assume that good estimates of λ_1 and $\mathbf{v}_1$ are known in the sense that $|\lambda_1 - \lambda|$ is small, and $|z_{1j}/z_{11}| \leq \gamma_1 |\lambda_1 - \lambda|$. Also that λ_1 is isolated in the sense that $|\lambda_1 - \lambda_j| \geq a,\ j = 2, \cdots, p$. Then the current iteration steps are defined by

$$\begin{aligned} \beta(\lambda) &= \frac{\kappa(\lambda_1 - \lambda)}{(\mathbf{s}^T B \mathbf{v}_1)\, z_{11} \left\{1 + (\lambda_1 - \lambda) \sum_{j=2}^{p} \frac{z_{1j}}{z_{11}} \frac{\mathbf{s}^T B \mathbf{v}_j}{\mathbf{s}^T B \mathbf{v}_1 (\lambda_j - \lambda)}\right\}}, \qquad (10) \\ \lambda^+ &= \lambda - \frac{\beta(\lambda)}{d\beta(\lambda)/d\lambda}, \\ B\mathbf{x}^+ &= \frac{B\mathbf{w}}{\|\mathbf{w}\|} \qquad (11) \end{aligned}$$

where updated quantities are indicated by a + superscript, and, in particular,

$$\left| \frac{z_{ij}^+}{z_{11}^+} \right| \leq \frac{\gamma_1}{a^2} |\lambda_1 - \lambda|^3 .$$

It follows from these developments that the iteration function $\beta(t)$ has the form

$$\beta(t) = \frac{\kappa\,(\lambda_1 - t)}{(1 + (\lambda_1 - t))\, g(\lambda)}, \qquad (12)$$

where $|g(t)| \leq \frac{\gamma_1}{a} |\lambda_1 - \lambda|$, $\left|g'(t)\right| \leq \frac{\gamma_1}{a^2} |\lambda_1 - \lambda|$ provided $|\lambda_j - t| \geq a$, and $\mathbf{s}^T B \mathbf{v}_j / \mathbf{s}^T B \mathbf{v}_1 \leq 1,\ j = 2, \cdots p$, as $\mathbf{x}, \mathbf{s}$ are considered fixed at the current iteration step. The general Newton step computed from (12) is

$$\frac{\beta(t)}{d\beta(t)/dt} = -\frac{\lambda_1 - t}{1 - \beta(t)\left((\lambda_1 - t) g(t)' - g(t)\right)}.$$

It follows that

$$\lambda_1 - t = \frac{\beta(t)}{d\beta(t)/dt} - \frac{\beta(t)^2}{d\beta(t)/dt} \left((\lambda_1 - t) g(t)' - g(t)\right). \qquad (13)$$

Thus, the correction to the current estimate obtained by setting $t = \lambda$ in (13) satisfies

$$\left| \lambda_1 - \lambda - \frac{\beta}{d\beta(\lambda)/d\lambda} \right| \leq \zeta |\lambda_1 - \lambda|^3 . \qquad (14)$$

Remark 1 *It is an immediate consequence of this form for the iteration function and the above estimates that the inverse iteration Newton method (4) coupled with the inverse iteration corrected* $\mathbf{x}$ *(11) has third order convergence for linear problems provided* $\mathbf{s}^T B \mathbf{v}_1 \neq 0$. *A suitable form for* $\mathbf{s}$ *is*

$$\mathbf{s} = \mathbf{e}_k,\ k = \arg \max_i |w_i| .$$

This gives an iteration which does not require factors of the transpose matrix.

Note that $\mathbf{s}$ can be updated in the same manner as $\mathbf{x}$. Set

$$\begin{aligned}
\mathbf{s} &= S\mathbf{t}, \\
\tilde{\mathbf{s}} &= (A-\lambda B)^{-T} B^T \mathbf{s} = S(\Lambda - \lambda I)^{-1}\mathbf{t}, \\
\mathbf{s}^+ &= (A-\lambda B)^{-T} B^T \tilde{\mathbf{s}} = S(\Lambda-\lambda I)^{-2}\mathbf{t} / \left\| S(\Lambda-\lambda I)^{-2}\mathbf{t} \right\|.
\end{aligned}$$

Then

$$\frac{t_j}{t_1} = O(|\lambda_1 - \lambda|) \rightarrow \frac{t_j^+}{t_1^+} = O(|\lambda_1-\lambda|^3).$$

In this case we have

$$\mathbf{s}^T B \mathbf{v}_j = O(|\lambda_1 - \lambda|),\ j = 2, \cdots, p,$$

giving the inequalities

$$|g(t)|\ , \left|g(t)^{'}\right| \leq O(|\lambda_1-\lambda|^2).$$

This suggests that an even faster rate of convergence is possible by also choosing $\mathbf{s}$ suitably. The situation is a little more complicated because there is no corresponding mechanism to improve the convergence rate of $\mathbf{x}, \mathbf{s}$ except by performing further inverse iteration steps. For example, one further step before updating of both $\mathbf{x}$ and $\mathbf{s}$ would yield a fourth order method. If no further steps are performed then the rates can be estimated by noting that if ε_k estimates the error in λ at the k'th step, and δ_k is the corresponding estimate for the errors in $\mathbf{x}$ and $\mathbf{s}$, then the orders satisfy

$$\begin{aligned}
\varepsilon_{k+1} &= \varepsilon_k^2 \delta_k^2, \\
\delta_{k+1} &= \varepsilon_k^2 \delta_k.
\end{aligned}$$

Eliminating between these equations gives the three term recurrences

$$\begin{aligned}
\varepsilon_{k+2} &= \varepsilon_{k+1}^3 \varepsilon_k^2, \\
\delta_{k+2} &= \delta_{k+1}^3 \delta_k^2.
\end{aligned}$$

The usual argument now shows that that the eigenvalue process has rate 3.56. These iterations which give a better than third order convergence rate require factors of the transpose matrix in order to update $\mathbf{s}$.

3 Deflation in the eigenvalue calculation

The idea of a deflation method is to move eigenvalues already computed out of the way while preserving those yet to be determined. This can be attempted either

- by dividing out in the iteration function to give a new function

$$\phi_k(\lambda) = \frac{\beta(\lambda)}{\prod_{i=1}^{k-1}(\lambda - \lambda_i)}; \tag{15}$$

- by modifying the matrix using existing spectral information.

Weilandt deflation [6], a version of the second approach, has proved most successful in linear problems. Consider

$$A_i^* = \left(I - \sigma B \mathbf{v}_i \mathbf{s}_i^T\right) A.$$

This maps $\lambda_i \leftarrow (1-\sigma)\lambda_i$, and leaves the remaining eigenvalues unchanged. In terms of the new matrix the basic iteration step becomes

$$\left(A_i^* - \lambda B\right) \mathbf{v} = \beta(\lambda) B \mathbf{x}.$$

This is equivalent to

$$\left(A - \lambda \left(I + \frac{\sigma}{1-\sigma} B \mathbf{v}_i \mathbf{s}_i^T\right) B\right) \mathbf{v} = \beta(\lambda) \left(I + \frac{\sigma}{1-\sigma} B \mathbf{v}_i \mathbf{s}_i^T\right) B \mathbf{x}.$$

Letting $\sigma \to \infty$ moves $\lambda_i \to \pm\infty$ which should be out of the way for all practical purposes. Define the (non-orthogonal) projection P_i by

$$P_i = I - B \mathbf{v}_i \mathbf{s}_i^T.$$

Then the limiting form of the basic iteration as $\sigma \to \infty$ can be written

$$\left(A - \lambda P_i B\right) \mathbf{v} = \beta(\lambda) P_i B \mathbf{x},$$

and the solution of this equation is the same (has the same Fourier coefficients) as that of the system

$$\left(A - \lambda B\right) \mathbf{v} = \beta(\lambda) P_i B \mathbf{x}.$$

The distinction is important as any sparsity in $(A - \lambda B)$ is preserved in this second system. Thus this is a procedure which would seem to be most appropriate for narrow banded (A, B) where the penalty for refining estimates of both left and right vectors is least.

If (A, B) is full then it is worth investigating alternative deflation methods that do not require some form of access to the transpose matrix as these could have advantages for implementation on distributed memory hardware. One

possibility is Schur-Weilandt deflation [6]. Let the eigenvector matrix be factored as

$$V = QZ,$$

where Q has orthonormal columns and $Z : C^p \rightarrow C^p$ is upper triangular. Then this gives a form of Schur decomposition

$$AQ = BQZ\Lambda Z^{-1} = BQR$$

where $R : C^p \rightarrow C^p$ is upper triangular, and it is convenient to refer to the columns of Q as Schur vectors. Set

$$Q_k = QE_k,\ E_k = [\mathbf{e}_1, \cdots, \mathbf{e}_k],$$

and define

$$A_k^* = A - \sigma BQ_kQ_k^H.$$

Then

$$A_k^*Q = BQR - \sigma BQE_kE_k^T.$$

This shows that A_k^* has the same Schur vectors as A, and eigenvalues

$$\begin{aligned} \lambda_j^k &= \lambda_j - \sigma,\ j = 1, \cdots, k, \\ \lambda_j^k &= \lambda_j,\ j = k+1, \cdots, p \end{aligned}$$

. Also, let

$$(A_k^* - \lambda_{k+1}B)\,\mathbf{v}_{k+1}^k = 0.$$

Then $\mathbf{v}_{k+1}^k \in \text{span}\,\{\mathbf{v}_1, \cdots, \mathbf{v}_{k+1}\}$, so that orthonormalising it to the columns of Q_k generates the next Schur vector in the sequence. This suggests that the A_k^*, $k = 1, 2, \cdots$ would form a suitable deflation sequence provided σ can be chosen appropriately. There are two problems with this suggestion:

1. The eigenvector information is obtained in the form of Schur vectors, so a post processing stage has to be used to obtain the eigenvectors;

2. In order to move the already computed eigenvalues away from any further contention it is necessary to choose $|\sigma|$ large enough. This could have some impact on the size of $\text{cond}(A)$.

Selection of σ can be made somewhat simpler by deflating the iteration function $\beta_k(\lambda)$ associated with A_k^* using the first deflation approach (15) to remove the shifted eigenvalues from contention. This suggests applying Newton's method to

$$\phi_k(\lambda) = \frac{\beta_k(\lambda)}{\prod_{j=1}^k (\lambda + \sigma - \lambda_j)}. \tag{16}$$

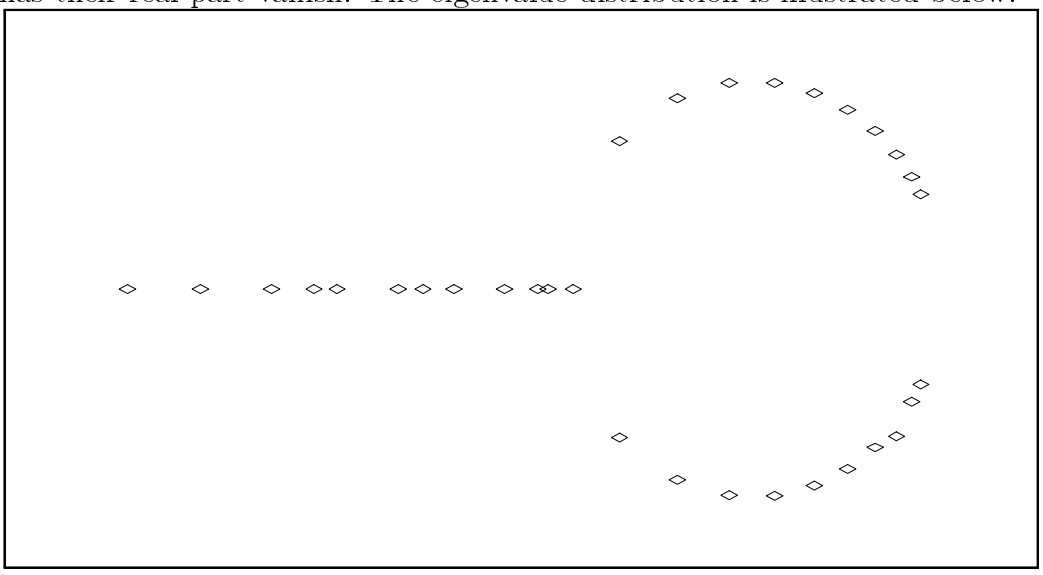
There is snag, however, as the rate of convergence is reduced from third order to second order if $\phi_k(\lambda)$ is used as the iteration function instead of $\beta_k(\lambda)$.

4 Summary of numerical results

Two main groups groups of numerical computations have been performed. The first has involved the testing of Weilandt deflation used in conjunction with a banded (block bidiagonal) linear equation solver. The computations have been carried out using two different discretizations of the brusselator equation based on the description given in [6]. This problem comprises a system of two coupled partial differential equations which possess a constant steady state solution. Examination of the stabitity of the steady state leads to consideration of the coupled ordinary differential equation eigenvalue problem:

$$\begin{aligned} \frac{D_X}{L^2}\frac{d^2X}{dt^2} + (\beta - 1)X + \alpha^2 Y &= \lambda X, \\ \frac{D_Y}{L^2}\frac{d^2Y}{dt^2} - \beta X - \alpha^2 Y &= \lambda Y. \end{aligned}$$

The problem of most interest is determining a Hopf bifurcation. This amounts to finding the value of L for which the right-most complex pair of eigenvalues has their real part vanish. The eigenvalue distribution is illustrated below:



Brusselator eigenvalues

Two different procedures are used to set up the block bidiagonal matrices.

(a) The differential equations are written as a first order system of order 4, and then integrated using the trapezoidal rule. This leads directly to a block bidiagonal system with 4×4 blocks. In this case the weighting matrix B does not have full rank; and

(b) The standard central difference formula is used to discretize the second order equations. If X and Y values associated with the same mesh point are stored in consecutive locations in the eigenvector then the result is block tridiagonal with 2×2 blocks. This becomes block bidiagonal by grouping pairs of adjacent blocks to again form 4×4 blocks.

Random initial conditions for the starting vectors, and an initial choice for the eigenvalue of a perturbation of the origin with a standard deviation of .1 have been used. This is a choice which favours finding eigenvalues of small magnitude initially in the brusselator problem, and this is just what happens. As the real parts of the eigenvalues become large negative it becomes a less favourable guess. Also, there is some uncertainty as to which of the two smallest eigenvalues is found first as the discretization interval and finite difference formulae are varied. This behaviour can be made more predictable by performing several steps of inverse iteration to emphasise the component of the eigenvector associated with the smallest eigenvalue on the first iteration for each new eigenvalue. The figures given here were obtained using 3 initial steps of inverse iteration. This works because the deflation process has banished the eigenvalues already found to ∞, and the figure shows why it is reasonable to expect that the computed eigenvalues will be approximately consistently ordered for the different methods used if the computations are started with an initial value near 0. Results have been similar for both the block bidiagonal formulations and prove generally satisfactory. The progress of inverse iteration implemented in conjunction with Weilandt deflation - the algorithm with rate 3.56 - is detailed in Table 1. This summarises the computation of the first, seventeenth, and thirty second eigenvalues for the second difference approximation. A mesh with $\Delta t = 1/41$ and a block bidiagonal matrix with 20 4×4 blocks were used. In this case the algorithm was asked to find 32 of a possible 80 eigenvalues. The corresponding mesh in the first order system gives 40 4×4 blocks. However, because B is singular the system has only 76 eigenvalues.

n	Re λ	Im λ	Re $\Delta\lambda$	Im $\Delta\lambda$
1	.2271+ 1	.2272+ 1	.2258+ 1	.2128+ 1
	-.1779+ 0	.2290+ 1	-.2449+ 1	.1868- 1
	-.9781- 3	.2145+ 1	.1769+ 0	-.1455+ 0
	.1101- 3	.2139+ 1	.1088- 2	-.5317- 2
	.1103- 3	.2139+ 1	.8339-12	.1319-10
17	-.2101+ 2	-.3282+ 0	-.2095+ 2	-.3673+ 0
	-.2954+ 2	-.2902+ 1	-.8533+ 1	-.2574+ 1
	-.2174+ 2	-.4393+ 1	.7806+ 1	-.1491+ 1
	-.2120+ 2	-.3652+ 1	.5325+ 0	.7406+ 0
	-.2120+ 2	-.3647+ 1	.8153- 2	.4741- 2
	-.2120+ 2	-.3647+ 1	-.6839-13	-.1395-12
32	-.4601+ 2	.3089+ 0	-.4593+ 2	.3753+ 0
	-.4604+ 2	-.2174- 4	-.3702- 1	-.3089+ 0
	-.4604+ 2	-.5353-18	-.1580- 4	.2174- 4
	-.4604+ 2	.3387-30	.9304-15	.5353-18

Table 1. Convergence of selected eigenvalues, Weilandt deflation.

The second set of numerical experiments has involved the testing of Schur-Weilandt deflation including the modification (16). Two types of test have been used in this application

a. Basic tests have been carried out using the matrices

$$A = \begin{bmatrix} 1.0 & 5.9 & 2.3 & 4.1 & 1.9 & .07 \\ .70 & 1.0 & 1.4 & 0.5 & 1.3 & 4.7 \\ 0.0 & .13 & 1.0 & 5.7 & 2.4 & 2.2 \\ 0.0 & 0.0 & .54 & 1.0 & .63 & 3.3 \\ 0.0 & 0.0 & 0.0 & .47 & 1.0 & 1.9 \\ 0.0 & 0.0 & 0.0 & 0.0 & .05 & 1.0 \end{bmatrix},$$

$$B = \begin{bmatrix} 1.0 & 5.9 & 2.3 & 4.1 & 1.9 & .07 \\ .70 & 1.0 & 1.4 & 0.5 & 1.3 & 4.7 \\ 0.0 & .13 & 1.0 & 5.7 & 2.4 & 2.2 \\ 0.0 & 0.0 & .54 & 1.0 & .63 & 3.3 \\ 0.0 & 0.0 & 0.0 & .47 & 1.0 & 1.9 \\ 7.4 & 0.0 & 0.0 & 0.0 & .05 & 1.0 \end{bmatrix},$$

$$C = \theta A + (1-\theta) B.$$

The eigenvalues of A are .71688, -1.0004, 1.1469, -1.4566, 2.9759, 3.6172, those of B are -1.7533$\pm$5.7511i, .84183$\pm$1.9336i, -1.7185, 9.5414. Both these problems have well conditioned eigenvalues, but θ can be chosen

so that C has repeated roots. For example, if $\theta = .9938404971$ then C has the eigenvalues $-.34179 \pm (.20917e-4)i$. For this value of θ, C is close to defective with a condition number of $O(1.e5)$.

b. The discretization of the brusselator problem in second order form provides an example where the deflation process can involve a larger number of eigenvalues. The same problem parameters as before are used.

The main interest in the small, well conditioned problems (test matrices A and B) proved to be in the choice of the shift parameter σ. This had to be chosen > 20, 25 being large enough for matrix A, and > 61, 63 being large enough in the case of matrix B, before the Schur-Weilandt routine would find all eigenvalues. The main interest in matrix C is in the ill-conditioned eigenvalue. There is evidence of slow convergence in this case (2-3 times as many iterations as in the well conditioned cases) . This might be expected because exactly defective eigenvalues give multiple roots in the iteration function $\beta(\lambda)$, and this leads to linear convergence in the Newton iteration.
In the Schur-Weilandt calculations involving the brusselator equation the aim has been to study the performance of the iteration for a range of values of the shift parameter σ. The values chosen were $iD_X/(L\Delta t)^2$ with $i = 5, 10, 20, 50, 100, 1000$. The values for $i = 1000$ showed some deviation from the largely consistent order obtained with the smaller values. Results for the other values are displayed in Table 2.

i	5	10	20	50	100
λ_1	8	8	8	8	8
λ_{17}	10	10	8	7	7
λ_{32}	13	10	8	6	5

Table 2. Numbers of iterations versus the shift parameter.

The smallest of the shift values is not too far from the Gerschgorin bound, so Table 2 would suggest that it can be important to choose the parameter large enough. There was little change in behaviour after $i = 50$. Convergence behaviour is illustrated in Table 3 where the analogous data to that of Table 1 is displayed. The shift parameter corresponds to $i = 50$.

n	Re λ	Im λ	Re $\Delta\lambda$	Im $\Delta\lambda$
1	.2420+ 0	-.9457- 1	-.3342+ 0	.7874- 1
	.5503+ 0	-.2432+ 0	-.3083+ 0	.1486+ 0
	.6956+ 0	-.8680+ 0	-.1452+ 0	.6248+ 0
	-.2864- 1	-.2612+ 1	.7242+ 0	.1744+ 1
	-.1605- 2	-.2143+ 1	-.2704- 1	-.4692+ 0
	.1101- 3	-.2139+1	-.1716- 2	-.3858- 2
	.1101- 3	-.2139+ 1	-.2372- 7	-.3000- 7
	.1101- 3	-.2139+ 1	.7551-16	-.3339-14
17	-.1990+2	.1455+ 1	.1991+ 2	-.1533+ 1
	-.2230+ 2	.2541+1	.2407+ 1	-.1086+ 1
	-.2144+ 2	.3085+ 1	-.8659+ 0	-.5438+ 0
	-.2120+ 2	.3650+ 1	-.2346+ 0	-.5651+ 0
	-.2120+ 2	.3647+ 1	-.7686- 2	.3106- 2
	-.2120+ 2	.3647+ 1	.3370- 6	-.3535- 6
	-.2120+ 2	.3647+ 1	.3584-13	.3471-14
32	-.3240+ 2	-.3054- 1	.3240+ 2	-.8117- 1
	-.4591+ 2	-.1683+ 0	.1351+ 2	.1377+ 0
	-.4761+ 2	-.1517- 1	.1700+ 1	-.1531+ 0
	-.4764+ 2	-.8683- 5	.2438- 1	-.1516- 1
	-.4764+ 2	-.8236-12	.3995- 5	-.8683- 5
	-.4764+ 2	.8297-14	.6545-12	-.8319-12

Table 3. Convergence of selected eigenvalues, Schur-Weilandt deflation.

. These results show that there is a significant advantage for the algorithm using Weilandt deflation. This is not only due to the difference in ultimate convergence rates, although the second order convergence to the higher eigenvalues consequent on the explicit deflation of the shifted eigenvalues is evident, but also there is evidence of a slower passage to the acceleration in the final stage of convergence.

As a comparison, Table 4 gives results obtained using a shift-invert Arnoldi method [6]. Displayed are the shifts employed and the Krylov subspace dimension required to compute a particular eigenvalue to 6 figures. In the first three cases the required eigenvalue was dominant in the shifted problem. In the last it was third in line and a rather larger Krylov subspace is required.

shift	Re λ	Im λ	dimension
0	.1101-4	.213944+1	11
-20	-.2119545+2	.3647177+1	12
-45	-.4604412+2	0.	7
-40	-.4604412+2	0.	17

Table 4. Computation of selected eigenvalues by shift-invert Arnoldi method.

Here the shift choice involved some a priori information, but this could also have been obtained by inspecting the subdominant eigenvalues in earlier steps.

References

[1] Andrew Cleary and M.R.Osborne, Eigenvalue solvers for large problems, in *Computational Techniques and Applications: CTAC93*, Editors David Stewart, Henry Gardner, and David Singleton, p. 140-147, World Scientific, 1994.

[2] D.C.Handscomb, Computation of the latent roots of a Hessenberg matrix by Bairstow's method, Computer J., 5, p. 139-141, 1962.

[3] Markus Hegland and M.R. Osborne, Wrap-around partitioning for block bidiagonal systems, Mathematics Research Report No. MRR 018-96, Australian National University. (http://wilton.anu.edu.au/research.reports/)

[4] M.R.Osborne, Inverse iteration, Newton's method, and nonlinear eigenvalue problems, in *The Contributions of J.H. Wilkinson to Numerical Analysis*, Symposium Proceeding Series No. 19, The Institute of Mathematics and its Applications, 1979.

[5] M.R.Osborne and Sidney Michaelson, The numerical solution of eigenvalue problems in which the eigenvalue parameter appears nonlinearly, with an application to differential eqations, Computer J., 7, p. 66-71, 1964.

[6] Y.Saad, *Numerical Methods for Large Eigenvalue Problems*, Halsted Press,1992.

[7] J.H.Wilkinson, The calculation of eigenvectors of codiagonal matrices, Computer J., 1, p. 90-96, 1958.

5 Appendix

Here an application of the inverse iteration Newton method is made to the problem of extracting quadratic factors from the characteristic polynomial of a real matrix A using real arithmetic.

5.1 Extraction of quadratic factors

The quadratic expression in the real matrix A is written

$$M(\alpha,\beta) = A^2 - 2\alpha A + \beta I. \tag{17}$$

It is a factor of the characteristic equation if

$$\alpha = \frac{\lambda_1+\lambda_2}{2},\ \beta = \lambda_1\lambda_2,\ \gamma = \alpha^2-\beta \geq 0,$$

where λ_1, λ_2 are real eigenvalues of A with multiplicity at least two if $\lambda_1 = \lambda_2$, or

$$\alpha = \mathrm{Re}\,\lambda,\ \beta = |\lambda|^2,\ \gamma = \alpha^2-\beta < 0,$$

if λ is a complex eigenvalue. If (α^*,β^*) define a quadratic factor of the characteristic polynomial of A, then the dimension of the corresponding null space of $M(\alpha^*,\beta^*)$ is at least two. If the dimension is exactly two then the elements of the null space are linear combinations of the corresponding eigenvectors $\mathbf{u}_i$, $i = 1,2$ if $\gamma \geq 0$, and linear combinations of $\mathrm{Re}\,\mathbf{z}$ and $\mathrm{Im}\,\mathbf{z}$ if $\gamma < 0$.

The basic idea in using this parametrization in conjunction with the inverse iteration strategy is to consider the problems

$$M(\alpha,\beta)\,\mathbf{v}_i = \phi_i(\alpha,\beta)\mathbf{x}_i, \tag{18}$$

$$\mathbf{s}_i^T\mathbf{v}_i = \kappa_i,\ i = 1,2. \tag{19}$$

Provided both the $\mathbf{x}_i$, and the $\mathbf{s}_i$ have non trivial projections into the required invariant subspaces and satisfy suitable linear independence properties then, letting $(\alpha,\beta) \to (\alpha^*,\beta^*)$, the previous argument shows that (α^*,β^*) satisfy the pair of equations

$$\phi_i(\alpha,\beta) = 0,\ i = 1,2. \tag{20}$$

Differentiating the defining relations and eliminating $\frac{\partial \mathbf{v}_i}{\partial\alpha}, \frac{\partial \mathbf{v}_i}{\partial\beta}$ gives

$$\frac{\partial\phi_i/\partial\alpha}{\phi_i} = \frac{\mathbf{s}_i^T M^{-1}\frac{\partial M}{\partial\alpha}\mathbf{v}_i}{\mathbf{s}_i^T\mathbf{v}_i},\ \frac{\partial\phi_i/\partial\beta}{\phi_i} = \frac{\mathbf{s}_i^T M^{-1}\frac{\partial M}{\partial\beta}\mathbf{v}_i}{\mathbf{s}_i^T\mathbf{v}_i}, i = 1,2.$$

This leads to the pair of equations for the Newton correction

$$\mathbf{s}_i^T\mathbf{v}_i + \mathbf{s}_i^T M^{-1}\frac{\partial M}{\partial \alpha}\mathbf{v}_i\delta\alpha + \mathbf{s}_i^T M^{-1}\frac{\partial M}{\partial \beta}\mathbf{v}_i\delta\beta = 0,\ i = 1, 2. \tag{21}$$

There is a corresponding equation for making a Newton correction in order to move the right vector estimate closer to the correct invariant subspace by expanding to first order the system

$$M(\alpha + \delta\alpha, \beta + \delta\beta)(\mathbf{v}_i + \delta\mathbf{v}_i) = 0.$$

This gives

$$\mathbf{v}_i + \delta\mathbf{v}_i = -M(\alpha, \beta)^{-1}\left(\frac{\partial M}{\partial \alpha}\mathbf{v}_i\delta\alpha + \frac{\partial M}{\partial \beta}\mathbf{v}_i\delta\beta\right),\ i = 1, 2. \tag{22}$$

This update is consistent with the derivation of (22). An application of Taylor's theorem, given ($\delta\alpha$, $\delta\beta$), provides a similar formula for updating the $\mathbf{s}_i$

$$\mathbf{s}_i + \delta\mathbf{s}_i = -M(\alpha, \beta)^{-T}\left(\frac{\partial M}{\partial \alpha}^T\mathbf{s}_i\delta\alpha + \frac{\partial M^T}{\partial \beta}\mathbf{s}_i\delta\beta\right),\ i = 1, 2. \tag{23}$$

To maintain independence there is no restriction in orthogonalising the columns of $V = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 \end{bmatrix}$, and fixing the $\mathbf{s}_i$ to be biorthogonal to them. This can be done conveniently by use of the singular value decomposition (svd) of V,

$$V = U_V\Sigma_V X^T.$$

The columns of U_V provide suitable vectors, and the components of Σ_V provide useful scaling information. Now let $S = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 \end{bmatrix}$, and consider the svd of

$$W = S^T U_V = Y\Sigma_S Z^T.$$

Provided the columns of U_V are orthonormal, a suitable biorthogonal set is given by

$$U_S \leftarrow SY\Sigma_S^{-1},\ U_V \leftarrow U_V Z. \tag{24}$$

5.2 Deflation techniques

Deflation can be achieved by dividing the iteration functions ϕ_i by a polynomial vanishing at the roots already found as in (15). Such a strategy is discussed in [2] in conjunction with a form of Bairstow's method. A form of Weilandt deflation is also available. Let

$$U_V = X_V W_V,\ U_S = X_S W_S$$

where X_V, X_S are matrices of a converged pair of right and left eigenvectors. Biorthogonality gives

$$\begin{aligned} U_S^T U_V &= I = W_S^T X_S^T X_V W_V, \\ &= W_S^T \begin{bmatrix} \mathbf{s}_1^T \mathbf{v}_1 & \\ & \mathbf{s}_2^T \mathbf{v}_2 \end{bmatrix} W_V. \end{aligned}$$

A key result is that A commutes with $U_V U_S^T$:

$$\begin{aligned} A U_V U_S^T &= A X_V W_V W_S^T X_S^T, \\ &= X_V \Lambda \begin{bmatrix} \mathbf{s}_1^T \mathbf{v}_1 & \\ & \mathbf{s}_2^T \mathbf{v}_2 \end{bmatrix} X_S^T, \\ &= X_V W_V W_S^T \Lambda X_S^T, \\ &= X_V W_V W_S^T X_S^T A = U_V U_S^T A. \end{aligned} \tag{25}$$

Define

$$D = I - \gamma U_V U_S^T, \; A^* = DA. \tag{26}$$

Then the eigenvalues corresponding to the invariant subspace defined by U_V are shifted by a factor $1 - \gamma$ and the remainder are unchanged. Making use of the above commutativity result gives

$$A^{*2} - 2\alpha A^* + \beta I = D^2 \left(A^2 - 2\alpha D^{-1} A + \beta D^{-2} \right).$$

The limiting form of the iteration equation as $\gamma \to \infty$ is

$$\left(A^2 - 2\alpha (I - U_V U_S^T) A + \beta (I - U_V U_S^T) \right) \mathbf{v} = \phi (I - U_V U_S^T) \mathbf{x}.$$

But this has the same solution as

$$\left(A^2 - 2\alpha A + \beta I \right) \mathbf{v} = \phi (I - U_V U_S^T) \mathbf{x}. \tag{27}$$

5.3 Numerical results

The main trouble with applying the quadratic factor extraction procedure would appear to be that it requires twice the amount of initial information to be supplied as in the case of single eigenvalue extraction, and this appears to increase the difficulty by a factor of at least 2**2! Already on the perturbed Hessenberg matrix B, while the procedure would function as expected with starting values for the eigenvalues correct to within 5-10%, it was distinctly unhappy with starting values obtained by making random small perturbations about the origin. The main problem here turned out to be the two real

eigenvalues which also happen to be the extreme values of the real parts of the spectrum. The procedure proved capable of latching onto one of the real values and converging to it and a completely spurious value. The mechanism by which this was achieved involved one of the singular values in the step which orthogonalised the basis vectors for the next iteration becoming very small. Despite a fair amount of effort, it did not prove possible to develop an acceptable compromise between reasonably general initialisation procedures and adequate convergence behaviour. Schur-Weilandt deflation and complex arithmetic used to extract a single eigenvalue at a time seems definitely more satisfactory.