A New Deflation Criterion for the QR Algorithm

Mario Ahues[*] Françoise Tisseur[†]

January 14, 1997

Abstract

We present a new deflation criterion for the multishift QR algorithm motivated by convergence analysis for the basic QR algorithm. The performance of the criterion is illustrated by numerical experiments using the LAPACK implementation of the double-shift QR algorithm.

1 Introduction

The practical QR algorithm computes the eigenvalues of a full matrix $A \in C^{n\times n}$ by first reducing A to upper Hessenberg form by a unitary similarity transformation and then iterating on the Hessenberg form to reduce it to upper triangular form. On each iteration a deflation test is used to determine whether a subdiagonal element can be neglected and the original problem replaced by two subproblems of smaller size. Let H denote an upper Hessenberg matrix QR iterate. If $h_{i,i-1}$ is "sufficiently small", for some $2 \le i \le n$, then we replace $h_{i,i-1}$ by zero, obtaining

$$\begin{array}{cc} \begin{pmatrix} H_{11} & H_{12} \\ 0 & H_{22} \end{pmatrix} & \begin{array}{c} {\scriptstyle i-1} \\ {\scriptstyle n-i+1} \end{array} \\ \begin{array}{cc} {\scriptstyle i-1} & {\scriptstyle n-i+1} \end{array} & \end{array} ;$$

we say that the problem decouples into two problems of smaller order involving H_{11} and H_{22}.

Let i be the largest integer such that $h_{i,i-1}$ is small enough to be set to zero. If $i = n$, then we have found an eigenvalue. If $i = n-1$, we have found two eigenvalues,

[*]Equipe d'Analyse Numérique, Faculté des Sciences et Techniques de St-Etienne, 23, rue du Dr Paul Michelon, 42023 St-Etienne, cedex 02, France, `ahues@anumsun1.univ-st-etienne.fr`.

[†]Department of Computer Science, Ayres Hall, Knoxville, Tennessee 37996-1301, U.S.A., `ftisseur@cs.utk.edu`

those of the trailing 2×2 principal submatrix. Otherwise we continue iterating with the submatrix corresponding to rows and columns from i to n.

We now clarify what we mean by "sufficiently small". Suppose that the elements of the original matrix A are known to a precision of the order $\mathbf{u}\|A\|_2$ where $\mathbf{u}$ is the unit roundoff. Denote the Hessenberg matrix at step k of the iteration by $H_k = (h_{ij}^{(k)})$. If

$$|h_{i,i-1}^{(k)}| \leq \mathbf{u}\|A\|_F \tag{1.1}$$

then, since A is unitarily similar to H_k, setting $h_{i,i-1}^{(k)}$ to zero corresponds to a perturbation in A that is of order $\mathbf{u}\|H\|_2$, which is a negligible perturbation.

With the use of the criterion (1.1), the QR algorithm is backward stable [6]. Stewart [5] notes, however, that this criterion is unsatisfactory for graded matrices of the form illustrated by

$$A = \begin{pmatrix} 1 & 10^{-2} & 10^{-4} & 10^{-6} \\ 10^{-2} & 10^{-4} & 10^{-6} & 10^{-8} \\ 10^{-4} & 10^{-6} & 10^{-8} & 10^{-10} \\ 10^{-6} & 10^{-8} & 10^{-10} & 10^{-12} \end{pmatrix},$$

for which some of the eigenvalues may be much smaller than $\mathbf{u}\|A\|_F$. Suppose that $h_{nn}^{(k)}$ is converging towards an eigenvalue that is smaller than $\mathbf{u}\|A\|_F$. The criterion (1.1) might set $h_{n,n-1}^{(k)}$ to zero when it is larger than $h_{nn}^{(k)}$, in which case $h_{nn}^{(k)}$ is not a good approximate eigenvalue in a relative sense. It therefore seems better to have a criterion that compares the subdiagonal elements to their neighboring diagonal elements.

Most of the practical implementations of the QR algorithm (see, e.g., [1], [4]) use a deflation criterion proposed by Wilkinson [2]. At iteration k, the element in position $(i, i-1)$ of the iterated Hessenberg matrix H is set to zero if

$$|h_{i,i-1}^{(k)}| \leq \mathbf{u}(|h_{i-1,i-1}^{(k)}| + |h_{i,i}^{(k)}|). \tag{1.2}$$

Note that if this criterion is satisfied then the more crude criterion (1.1) is automatically satisfied.

The criterion (1.2) is essentially a heuristic test motivated by consideration of graded matrices, and while many years of experience show that it performs well in practice, there appears to be no mathematical theory to support it.

In this paper, we suggest a new deflation criterion based on mathematical considerations. The ideas and results presented in this work were motivated by an elementary 2×2 example. If

$$H = \begin{pmatrix} \tilde{\lambda}_1 & M \\ \varepsilon & \tilde{\lambda}_2 \end{pmatrix},$$

where ε is very small but M is very large, it is easy to check that the true eigenvalues are close to the approximate ones $\tilde{\lambda}_1$ and $\tilde{\lambda}_2$ only if $\varepsilon M/|\tilde{\lambda}_1 - \tilde{\lambda}_2|$ is very small. This suggests that deflation should take into account simultaneously three parameters:

1) The size of the subdiagonal elements,

2) the size of the strictly upper-triangular elements, and

3) the distance between the two diagonal elements.

In the next section, we give an analysis that generalizes the 2×2 example. In Section 3 we then propose a deflation criterion for the implicit multishift QR algorithm. We illustrate the performance of the deflation criterion numerically using the LAPACK implementation of the double-shift QR algorithm.

2 Approximation by the Diagonal Coefficients

Consider an upper Hessenberg matrix of the form

$$H_n(\varepsilon) = \begin{pmatrix} h_{11} & \cdots & \cdots & h_{1n} \\ \eta_1(\varepsilon) & \ddots & & \vdots \\ & \ddots & \ddots & \vdots \\ 0 & & \eta_{n-1}(\varepsilon) & h_{nn} \end{pmatrix}$$

where the subdiagonal entries are functions of ε and satisfy $\eta_i(0) = 0$. The analysis in this section makes no reference to the QR iteration, but in the next section, we regard $H_n(\varepsilon)$ as a QR iterate.

Let $\lambda(\varepsilon)$ be an eigenvalue of $H_n(\varepsilon)$. If the function $\varepsilon \in \mathbb{C} \mapsto \lambda(\varepsilon) \in \mathbb{C}$ is analytic in a neighborhood of 0 and if $\lambda(0) = h_{ii}$, then a first order MacLaurin expansion gives the bound

$$|\lambda(\varepsilon) - h_{ii}| \;\leq\; |\lambda'(0)\;\varepsilon| + O(|\varepsilon|^2). \tag{2.1}$$

If some eigenvalues are defective, we need to increase the order of the expansion. Hence $|\lambda'(0)|$ measures the quality of h_{ii} as an approximate eigenvalue and can be used to suggest a stopping criterion for the basic QR algorithm.

In the following, we give two ways to obtain a simple expression for $\lambda'(0)$. The first one uses the characteristic polynomial of H_n and the second uses some results from perturbation theory.

2.1 Characteristic polynomial

We begin by giving some useful lemmas. The first concerns the characteristic polynomial of $H_n(\varepsilon)$.

Lemma 1 *For $n > 2$, the characteristic polynomial of H_n, denoted by $p_n(\varepsilon, \cdot)$, is given by:*

$$\begin{aligned} p_n(\varepsilon, x) \;&=\; (h_{nn} - x)p_{n-1}(\varepsilon, x) - \eta_{n-1}(\varepsilon)\; h_{n-1,n}\; p_{n-2}(\varepsilon, x) \\ &\qquad + \eta_{n-1}(\varepsilon)\eta_{n-2}(\varepsilon)\Delta_{n-2}(\varepsilon, x), \end{aligned}$$

where $\Delta_{n-2}(\varepsilon, x)$ is the determinant of a matrix of order $n-2$ depending on ε and x.

Proof. If we expand $\det(H_n(\varepsilon) - xI)$ with respect to the last row, we obtain

$$p_n(\varepsilon, x) = (h_{nn} - x)p_{n-1}(\varepsilon, x) - \eta_{n-1}(\varepsilon)q_{n-1}(\varepsilon, x) \tag{2.2}$$

where $q_{n-1}(\varepsilon, x)$ is defined by

$$q_{n-1}(\varepsilon, x) = \det \begin{pmatrix} h_{11} - x & \dots & \dots & h_{1n} \\ \eta_1(\varepsilon) & \ddots & & \vdots \\ & \ddots & h_{n-2,n-2} - x & h_{n-2,n} \\ 0 & & \eta_{n-2}(\varepsilon) & h_{n-1,n} \end{pmatrix}.$$

In a similar way, expanding $q_{n-1}(\varepsilon, x)$ with respect to the last row, yields

$$q_{n-1}(\varepsilon, x) = h_{n-1,n}p_{n-2}(\varepsilon, x) - \eta_{n-2}(x)\Delta_{n-2}(\varepsilon, x), \tag{2.3}$$

where $\Delta_{n-2}(\varepsilon, x)$ is defined by

$$\Delta_{n-2}(\varepsilon, x) = \det \begin{pmatrix} h_{11} - x & \dots & \dots & h_{1n} \\ \eta_1(\varepsilon) & \ddots & & \vdots \\ & \ddots & h_{n-3,n-3} - x & h_{n-3,n} \\ 0 & & \eta_{n-3}(\varepsilon) & h_{n-2,n} \end{pmatrix}.$$

□

Because $p_n(\varepsilon, \lambda(\varepsilon))$ is identically zero for all sufficiently small ε,

$$\frac{d}{d\varepsilon}p_n(\varepsilon, \lambda(\varepsilon)) = \frac{\partial p_n}{\partial \varepsilon}(\varepsilon, \lambda(\varepsilon)) + \lambda'(\varepsilon)\frac{\partial p_n}{\partial x}(\varepsilon, \lambda(\varepsilon)) = 0.$$

The last equality applied to $\lambda(0) = h_{ii}$ leads to

$$\lambda'(0)\frac{\partial p_n}{\partial x}(0, h_{ii}) = -\frac{\partial p_n}{\partial \varepsilon}(0, h_{ii}). \tag{2.4}$$

The next lemma concerns the calculation of $\frac{\partial p_n}{\partial x}(0, h_{ii})$ and $\frac{\partial p_n}{\partial \varepsilon}(0, h_{ii})$.

Lemma 2 *Suppose that the η_i are differentiable in GGGGa neighborhood of 0. For $n > 1$, we have*

$$\frac{\partial p_n}{\partial \varepsilon}(0, x) = -\sum_{i=1}^{n-1} \eta_i^{'}(0)\ h_{i,i+1} \left\{ \prod_{\substack{j=1 \\ j \neq i, i+1}}^{n} (h_{jj} - x) \right\}, \tag{2.5}$$

Proof.

The proof is by induction. If $\eta_1^{'}(0)$ is defined, then, for $n=2$,

$$\frac{\partial p_2}{\partial \varepsilon}(0,x) = -\eta_1^{'}(0) h_{12}.$$

Using Lemma 1, for $n \geq 3$, we have

$$p_n(\varepsilon,x) = (h_{nn}-x)p_{n-1}(\varepsilon,x) - \eta_{n-1}(\varepsilon)\ h_{n-1,n}\ p_{n-2}(\varepsilon,x) + \eta_{n-1}(\varepsilon)\eta_{n-2}(\varepsilon)\mathit{\Delta}_{n-2}(\varepsilon,x)$$

Since the functions η_i are differentiable in a neighborhood of 0,

$$\begin{aligned}\frac{\partial p_n}{\partial \varepsilon}(\varepsilon,x) &= (h_{nn}-x)\frac{\partial p_{n-1}}{\partial \varepsilon}(\varepsilon,x) - \eta_{n-1}^{'}(\varepsilon)\ h_{n-1,n}p_{n-2}(\varepsilon,x) - \eta_{n-1}(\varepsilon)\ h_{n-1,n}\frac{\partial p_{n-2}}{\partial \varepsilon}(\varepsilon,x) \\ &\quad + \eta_{n-1}^{'}(\varepsilon)\eta_{n-2}(\varepsilon)\mathit{\Delta}_{n-2}(\varepsilon,x) + \eta_{n-1}(\varepsilon)\eta_{n-2}^{'}(\varepsilon)\mathit{\Delta}_{n-2}(\varepsilon,x) \\ &\quad + \eta_{n-1}(\varepsilon)\eta_{n-2}(\varepsilon)\frac{\partial \mathit{\Delta}_{n-2}}{\partial \varepsilon}(\varepsilon,x).\end{aligned}$$

Since for $n \geq 1$,

$$p_n(0,x) = \prod_{j=1}^{n}(h_{jj}-x), \tag{2.6}$$

and $\eta_i(0)=0$, evaluating $\frac{\partial p_n}{\partial \varepsilon}(\varepsilon,x)$ in $\varepsilon=0$ leads to

$$\begin{aligned}\frac{\partial p_n}{\partial \varepsilon}(0,x) &= -(h_{nn}-x)\sum_{i=1}^{n-2}\eta_i^{'}(0)h_{i,i+1}\left\{\prod_{\substack{j=1\\ j\neq i,i+1}}^{n-1}(h_{jj}-x)\right\} \\ &\quad - \eta_{n-1}^{'}(0)h_{n-1,n}\prod_{j=1}^{n-2}(h_{jj}-x) \\ &= -\sum_{i=1}^{n-1}\eta_i^{'}(0)\ h_{i,i+1}\left\{\prod_{\substack{j=1\\ j\neq i,i+1}}^{n}(h_{jj}-h_{ii})\right\},\end{aligned}$$

as claimed. $\square$

Lemma 3 *For $n \geq 1$,*

$$\frac{\partial p_n}{\partial x}(0,h_{ii}) = -\prod_{\substack{j=1\\ j\neq i}}^{n}(h_{jj}-h_{ii}). \tag{2.7}$$

Proof. If we differentiate (2.6) with respect to x, we obtain

$$p_n'(0,x) = -\sum_{k=1}^{n} \prod_{\substack{j=1 \\ j\neq i}}^{n} (h_{jj} - x),$$

and the result follows with $x = h_{ii}$. $\square$

The next theorem gives a simple expression for $\lambda^{'}(0)$.

Theorem 1 *Assume that $\lambda'(0)$ exists, the $\eta_i(.)$ are differentiable in a neighborhood of* 0 *and that the h_{ii} are all distinct. Then,*

If $\lambda(0) = h_{11}$, $\quad \lambda'(0) = \dfrac{\eta_1'(0)h_{12}}{h_{11} - h_{22}}.$

If $\lambda(0) = h_{nn}$, $\quad \lambda'(0) = \dfrac{\eta_{n-1}'(0)h_{n-1,n}}{h_{n-1,n-1} - h_{nn}}.$

If $\lambda(0) = h_{ii}$ *for an* i *such that* $2 \leq i \leq n-1$,

$$\lambda'(0) = \frac{\eta_{i-1}'(0)h_{i-1,i}}{h_{i,i} - h_{i-1,i-1}} + \frac{\eta_i'(0)h_{i,i+1}}{h_{i,i} - h_{i+1,i+1}}.$$

Proof. If the h_{ii} are all distinct then, following (2.7), $\dfrac{\partial p_n}{\partial x}(0, h_{ii})$ is nonzero. If the η_i are differentiable at 0, equations (2.4) and (2.5) leads to

$$\begin{aligned}
\lambda'(0) &= \frac{-\dfrac{\partial p_n}{\partial \varepsilon}(0, h_{ii})}{\dfrac{\partial p_n}{\partial x}(0, h_{ii})} \\
&= \frac{\displaystyle\sum_{i=1}^{n-1} \eta_i^{'}(0)\ h_{i,i+1} \left\{ \prod_{\substack{j=1 \\ j\neq i,i+1}}^{n} (h_{jj} - h_{ii}) \right\}}{\displaystyle\sum_{i=1}^{n} \prod_{\substack{j=1 \\ j\neq i}}^{n} (h_{jj} - h_{ii})}.
\end{aligned}$$

After simplifications, we obtain the desired results. $\square$

2.2 Perturbation Theory

Here, we use classical results from function theory. We still suppose that the $\eta_i(\varepsilon)$ are differentiable functions in a neighborhood of zero. Let us rewrite $H_n(\varepsilon)$ in the form

$H_n(\varepsilon) = H_n(0) + \varepsilon E(\varepsilon)$ where

$$E(\varepsilon) = \begin{pmatrix} 0 & \cdots & \cdots & 0 \\ \eta_1'(\varepsilon) + O(\varepsilon^2) & \ddots & & \vdots \\ & \ddots & & \vdots \\ 0 & & \eta_{n-1}'(\varepsilon) + O(\varepsilon^2) & 0 \end{pmatrix}.$$

Suppose that λ is a simple eigenvalue of $H_n(\varepsilon) \in \mathbb{C}^{n\times n}$ and that x and y satisfy $H_n(0)x = \lambda x$ and $y^* H_n(0) = \lambda y^*$ with $\|x\|_2 = \|y\|_2 = 1$.

It can be shown that in a neighborhood of zero, there exist differentiable $x(\varepsilon)$ and $\lambda(\varepsilon)$ such that

$$H_n(\varepsilon)x(\varepsilon) = \lambda(\varepsilon)x(\varepsilon),$$

where $x(0) = x$ and $\lambda(0) = \lambda$.

By differentiating this equation with respect to ε and setting $\varepsilon = 0$ in the result, we obtain

$$H_n(0)x'(0) + E(0)x = \lambda'(0)x + \lambda x'(0).$$

Applying y^* to both side yields to

$$y^* E(0)x = \lambda'(0)y^* x.$$

As λ is a simple eigenvalue then $y^* x \neq 0$ and $\lambda'(0)$ is given by

$$\lambda'(0) = \frac{y^* E(0)x}{y^* x}. \tag{2.8}$$

Lemma 4 *Suppose that $\lambda = h_{ii}$ and $h_{ii} \neq h_{jj}, j \neq i$. Then x and y are defined by*

$$\begin{cases} x_k = -\frac{1}{h_{kk}-h_{ii}} \sum_{j=k+1}^{i} h_{kj}x_j, \ \textit{for } k > i, \\ x_k = 0 \ \textit{for } k < i \end{cases} \qquad \begin{cases} y_k = 0 \ \textit{for } k > i, \\ y_k = -\frac{1}{h_{kk}-h_{ii}} \sum_{j=k+1}^{i} h_{kj}x_j, \ \textit{for } k < i. \end{cases}$$

x_i and y_i are chosen such that $\|x\|_2 = \|y\|_2 = 1$.

Proof. The left and right eigenvectors x and y are the solutions of the upper and lower triangular systems $(H_n(0) - h_{ii}I)x = 0$ and $(H_n(0) - h_{ii}I)^* y = 0$. $\square$

Now, let us evaluate $\lambda'(0)$ using (2.8). First we have

$$E(0)x = \begin{pmatrix} 0 \\ \eta_1'(0)x_1 \\ \vdots \\ \eta_{i-1}'(0)x_{i-1} \\ \eta_i'(0)x_i \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad \text{and} \quad y = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ y_i \\ y_{i+1} \\ \vdots \\ y_n \end{pmatrix},$$

so that, $y^*E(0)x = \sum_{k=1}^{n-1} \eta'_k(0) x_k y_{k+1}$.
If $1 < i < n$, then

$$\begin{aligned} y^*E(0)x &= \eta'_{i-1}(0)x_{i-1}y_i + \eta'_i(0)x_i y_{i+1} \\ &= -\eta'_{i-1}(0)\frac{h_{i-1,i}}{h_{i-1,i-1} - h_{ii}} x_i y_i - \eta'_i(0)\frac{h_{i+1,i}}{h_{i+1,i+1} - h_{ii}} x_i y_i, \end{aligned}$$

and $y^*x = x_i y_i$. Finally,

$$\lambda'(0) = \eta'_{i-1}(0)\frac{h_{i-1,i}}{h_{ii} - h_{i-1,i-1}} + \eta'_i(0)\frac{h_{i+1,i}}{h_{ii} - h_{i+1,i+1}}.$$

If $i = 1$, $y^*E(0)x = \eta'_1(0)y_2$ and then

$$\lambda'(0) = \eta'_1(0)\frac{h_{12}}{h_{11} - h_{22}}.$$

If $i = n$, $y^*E(0)x = \eta'_{n-1}(0)x_{n-1}$ and then

$$\lambda'(0) = \eta'_{n-1}(0)\frac{h_{n-1,n}}{h_{n+1,n+1} - h_{nn}}.$$

There are the same expressions as in Theorem 1.

The next theorem gives a bound for the distance between the diagonal elements of $H_n(\varepsilon)$ and its eigenvalues.

Theorem 2 *With same assumptions as in Theorem* 1 *and the convention* $h_{00} = -\infty$, $h_{n+1,n+1} = +\infty$, *we have for* $i = 1, \ldots, n$,

$$|\lambda(\varepsilon) - h_{ii}| \leq \frac{|\eta_{i-1}(\varepsilon)|\ |h_{i-1,i}|}{|h_{i,i} - h_{i-1,i-1}|} + \frac{|\eta_i(\varepsilon)|\ |h_{i,i+1}|}{|h_{i,i} - h_{i+1,i+1}|} + O(|\varepsilon|^2). \tag{2.9}$$

Proof. We obtain (2.9) by using inequality (2.1) and applying Theorem 1 with

$$\varepsilon\eta'_i(0) = \eta_i(\varepsilon) + O(|\varepsilon|^2).$$

□

We conclude, as in Section 1, that the accuracy of the diagonal elements as eigenvalue approximations depends on three parameters:

1) The size of the subdiagonal elements,

2) the size of the offdiagonal elements, and

3) the distance between two successive diagonal elements.

Note that, in practice, the bound (2.9) can be easily evaluated to first order, since all the quantities involved are known at each QR iteration.

3 Deflation Criterion

In this section we propose a deflation criterion for the QR algorithm. We aim to neglect a subdiagonal element only if the adjacent diagonal element is sufficiently close to an eigenvalue of the current matrix. That is, at step k, we aim to neglect $h_{i,i-1}^{(k)}$ if

$$\frac{|\lambda(\varepsilon) - h_{ii}^{(k)}|}{|h_{ii}^{(k)}|} \leq \mathbf{u}.$$

Using Theorem 2 and approximating the upper bound in (2.9) by the first term, leads to the criterion

$$|h_{i,i-1}^{(k)}|\ |h_{i-1,i}^{(k)}| \leq \mathbf{u}|h_{ii}^{(k)}||h_{ii}^{(k)} - h_{i-1,i-1}^{(k)}| \tag{3.1}$$

(recall that $\eta_i(\varepsilon) = h_{i+1,i}^{(k)}$). Note that the term $h_{ii}^{(k)}$ on the right-hand side makes this test appropriate for graded matrices. Our deflation criterion is that both (3.1) and (1.1) are satisfied, where the latter condition is imposed so as to ensure that backward stability of the algorithm is maintained.

This criterion can be easily implemented in LAPACK [1] by modifying the routine `xLAHQR`. This routine is an implementation of the implicit double shift QR algorithm (see [2], [3], [4]).

After the lines

```
*
*     Perform QR iterations on rows and columns ILO to I until a
*     submatrix of order 1 or 2 splits off at the bottom because a
*     subdiagonal element has become negligible.
*
      DO 130 ITS = 0, ITN
*
*        Look for a single small subdiagonal element.
*
```

replace the lines

```
*
      DO 130 ITS = 0, ITN
*
*        Look for a single small subdiagonal element.
*
         DO 20 K = I, L + 1, -1
            TST1 = ABS( H( K-1, K-1 ) ) + ABS( H( K, K ) )
            IF( TST1.EQ.ZERO )
     $         TST1 = DLANHS( '1', I-L+1, H( L, L ), LDH, WORK )
            IF( ABS( H( K, K-1 ) ).LE.MAX( ULP*TST1, SMLNUM ) )
```

```
     $           GO TO 30
   20    CONTINUE
   30    CONTINUE
         L = K
         IF( L.GT.ILO ) THEN
*
*           H(L,L-1) is negligible
*
            H( L, L-1 ) = ZERO
         END IF
```

by the lines

```
*
      DO 130 ITS = 0, ITN
*
*        Look for a single small subdiagonal element.
*
         DO 20 K = I, L + 1, -1
            IF( ABS( H( K, K-1 )).GT. (ULP*NORM)) GO TO 20
            TST1 = ABS( H( K-1, K-1 ) - H( K, K ) )*
     $            ABS( H( K, K ) )
            SS  =  ABS( H( K-1, K ) )
            IF( SS.EQ.ZERO )  GO TO 30
            IF( ABS( H( K, K-1 )).LE.MAX( ULP*TST1/SS,SMLNUM ))
     $          GO TO 30
   20    CONTINUE
   30    CONTINUE
         L = K
         IF( L.GT.ILO ) THEN
*
*           H(L,L-1) is negligible
*
            H( L, L-1 ) = ZERO
         END IF
```

where `NORM` is the 1-norm of the matrix H and `ULP` is the machine precision.

Here is a small example where the new deflation criterion enhances the accuracy. Let H be defined by

$$H = \begin{pmatrix} 1 & M & 0 \\ \varepsilon & 1+d & M \\ 0 & \varepsilon & 1+2d \end{pmatrix}.$$

For $\varepsilon = 1.1e-8$, $M = 1.1e5$, $d = 1.e-2$, the classical test of `SLAHQR` (single precision) deflates immediately. The resulting relative error on the computed eigenvalues is of the order of $0.4e-1$. With the new stopping criterion, three more QR iterations are needed. The relative error obtained is of the order of $0.7e-7$.

We have made some experiments on several matrices. We have used the LAPACK routine `xDLATME` to generate random nonsymmetric square matrices with specified eigenvalues.

The two deflation tests are very similar concerning the number of iterations. The new criterion seems to require a few less iterations when the matrix is diagonalizable and a few more iterations when the matrix is non-normal (see Table 4.1).

In order to compare the precision of the computed eigenvalues, we ran the two tests in single precision and compared the results with those obtained in double precision with `DLAHQR` (see Table 4.2). Sometimes, the new test requires more iterations but gives a better accuracy and sometimes it requires less iterations and gives the same accuracy.

We also have tested graded matrices. As expected, the results are very similar to those obtained with the classical criterion concerning precision and number of iterations (see Table 4.3).

4 Conclusion and Suggestions

Our analysis suggests that a deflation criterion for the practical QR algorithm should take into account the size of the subdiagonal elements, the size of the strictly upper-triangular elements and the distance between consecutive diagonal elements. The criterion that we have proposed has these properties and our numerical experiments indicate that it is competitive with the traditional test (which is heuristically motivated) in practice.

It is surprising that the new criterion performs so similarly to the traditional one, while being very different in form. Our work raises the question of what is the best deflation criterion for the QR algorithm. Further work is needed to answer this question, but our work as shown that the heuristically motivated test used in LAPACK does not always lead to best relative error.

Aknowledgments

We thank David Day, Nick Higham for helpfull comments and Pete Stewart for suggesting the approach uses in section 2.2.

	Diagonalisable		Non normal	
n	Classical test	New test	Classical test	New test
20	43	44	46	46
40	96	91	103	104
60	144	140	163	166
80	174	173	200	198
100	211	210	251	248

Table 4.1: Matrices built using `DLATME` with MODE=0, D is set to random number in the range $[-1, 1]$. Tests done in double precision.

	Classical test		New test	
n	Nb. of iterations	Relative error	Nb. of iterations	Relative error
20	50	0.79D-01	51	0.79D-01
40	83	0.11D+01	86	0.41D+00
60	127	0.20D+01	124	0.20D+01

Table 4.2: Matrices built using `DLATME` with MODE=1 ($D(1) = 1, D(2 : n) = 10^{-2}$), CONDS=1.0D+02.

	Classical test		New test	
n	Nb. of iterations	Relative error	Nb. of iterations	Relative error
4	4	0.11D-03	4	0.11D-03
8	10	0.76D-02	10	0.76D-02
10	15	0.58D-02	19	0.58D-02
12	17	0.13D-02	21	0.13D-02

Table 4.3: Graded matrices.

References

[1] E. Anderson, Z. Bai, C. H. Bischof, J. W. Demmel, J. J. Dongarra, J. J. Du Croz, A. Greenbaum, S. J. Hammarling, A. McKenney, S. Ostrouchov, and D. C. Sorensen. *LAPACK Users' Guide, Release 2.0.* Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, second edition, 1995.

[2] R. S. Martin, G. Peters, and J. H. Wilkinson. The QR algorithm for real Hessenberg matrices. *Numer. Math.*, 14:219–231, 1970.

[3] G. Peters and J. H. Wilkinson. Eigenvectors of real and complex matrices by LR and QR triangularization. *Numerische Mathematik*, 16:181–204, 1970.

[4] B. T. Smith, J. M. Boyle, J. J. Dongarra, B. S. Garbow, Y. Ikebe, V. C. Klema, and C. B. Moler. *Matrix Eigensystem Routines - EISPACK Guide*, volume 6 of *Lecture Notes in Computer Sciences.* Springer-Verlag, Berlin, 1976.

[5] G. W. Stewart. *Introduction to Matrix Computations.* Academic Press, New York, 1973.

[6] Françoise Tisseur. Backward stability of the QR algorithm. Technical Report 239, Equipe d'Analyse Numerique, Université Jean Monnet de Saint-Etienne, Cedex 02, France, October 1996.