5.3 The Power Method

The *power method* is very good at approximating the *extremal* eigenvalues of the matrix, that is, the eigenvalues having largest and smallest module, denoted by λ_1 and λ_n respectively, as well as their associated eigenvectors.

Solving such a problem is of great interest in several real-life applications (geosysmic, machine and structural vibrations, electric network analysis, quantum mechanics,...) where the computation of λ_n (and its associated eigenvector $\mathbf{x}_n$) arises in the determination of the *proper frequency* (and the corresponding *fundamental mode*) of a given physical system. We shall come back to this point in Section 5.12.

Having approximations of λ_1 and λ_n can also be useful in the analysis of numerical methods. For instance, if A is symmetric and positive definite, one can compute the optimal value of the acceleration parameter of the Richardson method and estimate its error reducing factor (see Chapter 4), as well as perform the stability analysis of discretization methods for systems of ordinary differential equations (see Chapter 11).

5.3.1 Approximation of the Eigenvalue of Largest Module

Let $\mathrm{A} \in \mathbb{C}^{n\times n}$ be a diagonalizable matrix and let $\mathrm{X} \in \mathbb{C}^{n\times n}$ be the matrix of its right eigenvectors $\mathbf{x}_i$, for $i = 1, \ldots, n$. Let us also suppose that the eigenvalues of A are ordered as

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \ldots \geq |\lambda_n|, \tag{5.16}$$

where λ_1 has algebraic multiplicity equal to 1. Under these assumptions, λ_1 is called the *dominant* eigenvalue of matrix A.
Given an arbitrary initial vector $\mathbf{q}^{(0)} \in \mathbb{C}^n$ of unit Euclidean norm, consider for $k = 1, 2, \ldots$ the following iteration based on the computation of powers of matrices, commonly known as the *power method*:

$$\begin{aligned} \mathbf{z}^{(k)} &= \mathrm{A}\mathbf{q}^{(k-1)}, \\ \mathbf{q}^{(k)} &= \mathbf{z}^{(k)}/\|\mathbf{z}^{(k)}\|_2, \\ \nu^{(k)} &= (\mathbf{q}^{(k)})^H \mathrm{A}\mathbf{q}^{(k)}. \end{aligned} \tag{5.17}$$

Let us analyze the convergence properties of method (5.17). By induction on k one can check that

$$\mathbf{q}^{(k)} = \frac{\mathrm{A}^k\mathbf{q}^{(0)}}{\|\mathrm{A}^k\mathbf{q}^{(0)}\|_2}, \qquad k \geq 1. \tag{5.18}$$

This relation explains the role played by the powers of A in the method. Because A is diagonalizable, its eigenvectors $\mathbf{x}_i$ form a basis of $\mathbb{C}^n$; it is thus possible to represent $\mathbf{q}^{(0)}$ as

$$\mathbf{q}^{(0)} = \sum_{i=1}^{n} \alpha_i \mathbf{x}_i, \qquad \alpha_i \in \mathbb{C}, \qquad i = 1, \ldots, n. \tag{5.19}$$

Moreover, since $\mathrm{A}\mathbf{x}_i = \lambda_i \mathbf{x}_i$, we have

$$\mathrm{A}^k \mathbf{q}^{(0)} = \alpha_1 \lambda_1^k \left(\mathbf{x}_1 + \sum_{i=2}^{n} \frac{\alpha_i}{\alpha_1} \left(\frac{\lambda_i}{\lambda_1} \right)^k \mathbf{x}_i \right), \; k = 1, 2, \ldots \tag{5.20}$$

Since $|\lambda_i/\lambda_1| < 1$ for $i = 2, \ldots, n$, as k increases the vector $\mathrm{A}^k \mathbf{q}^{(0)}$ (and thus also $\mathbf{q}^{(k)}$, due to (5.18)), tends to assume an increasingly significant component in the direction of the eigenvector $\mathbf{x}_1$, while its components in the other directions $\mathbf{x}_j$ decrease. Using (5.18) and (5.20), we get

$$\mathbf{q}^{(k)} = \frac{\alpha_1 \lambda_1^k (\mathbf{x}_1 + \mathbf{y}^{(k)})}{\|\alpha_1 \lambda_1^k (\mathbf{x}_1 + \mathbf{y}^{(k)})\|_2} = \mu_k \frac{\mathbf{x}_1 + \mathbf{y}^{(k)}}{\|\mathbf{x}_1 + \mathbf{y}^{(k)}\|_2},$$

where μ_k is the sign of $\alpha_1 \lambda_1^k$ and $\mathbf{y}^{(k)}$ denotes a vector that vanishes as $k \to \infty$.

As $k \to \infty$, the vector $\mathbf{q}^{(k)}$ thus aligns itself along the direction of eigenvector $\mathbf{x}_1$, and the following error estimate holds at each step k.

Theorem 5.6 *Let $\mathrm{A} \in \mathbb{C}^{n \times n}$ be a diagonalizable matrix whose eigenvalues satisfy* (5.16)*. Assuming that $\alpha_1 \neq 0$, there exists a constant $C > 0$ such that*

$$\|\tilde{\mathbf{q}}^{(k)} - \mathbf{x}_1\|_2 \leq C \left| \frac{\lambda_2}{\lambda_1} \right|^k, \qquad k \geq 1, \tag{5.21}$$

where

$$\tilde{\mathbf{q}}^{(k)} = \frac{\mathbf{q}^{(k)} \|\mathrm{A}^k \mathbf{q}^{(0)}\|_2}{\alpha_1 \lambda_1^k} = \mathbf{x}_1 + \sum_{i=2}^{n} \frac{\alpha_i}{\alpha_1} \left(\frac{\lambda_i}{\lambda_1} \right)^k \mathbf{x}_i, \qquad k = 1, 2, \ldots \tag{5.22}$$

Proof. Since A is diagonalizable, without losing generality, we can pick up the nonsingular matrix X in such a way that its columns have unit Euclidean length, that is $\|\mathbf{x}_i\|_2 = 1$ for $i = 1, \ldots, n$. From (5.20) it thus follows that

$$\begin{aligned} \|\mathbf{x}_1 + \sum_{i=2}^{n} \left[\frac{\alpha_i}{\alpha_1} \left(\frac{\lambda_i}{\lambda_1} \right)^k \mathbf{x}_i \right] - \mathbf{x}_1\|_2 &= \|\sum_{i=2}^{n} \frac{\alpha_i}{\alpha_1} \left(\frac{\lambda_i}{\lambda_1} \right)^k \mathbf{x}_i\|_2 \\ \leq \left(\sum_{i=2}^{n} \left[\frac{\alpha_i}{\alpha_1} \right]^2 \left[\frac{\lambda_i}{\lambda_1} \right]^{2k} \right)^{1/2} &\leq \left| \frac{\lambda_2}{\lambda_1} \right|^k \left(\sum_{i=2}^{n} \left[\frac{\alpha_i}{\alpha_1} \right]^2 \right)^{1/2}, \end{aligned}$$

that is (5.21) with $C = \left(\sum_{i=2}^{n} (\alpha_i/\alpha_1)^2 \right)^{1/2}$. ◇

Estimate (5.21) expresses the convergence of the sequence $\tilde{\mathbf{q}}^{(k)}$ towards $\mathbf{x}_1$. Therefore the sequence of Rayleigh quotients

$$((\tilde{\mathbf{q}}^{(k)})^H \mathrm{A}\tilde{\mathbf{q}}^{(k)})/\|\tilde{\mathbf{q}}^{(k)}\|_2^2 = \left(\mathbf{q}^{(k)}\right)^H \mathrm{A}\mathbf{q}^{(k)} = \nu^{(k)}$$

will converge to λ_1. As a consequence, $\lim_{k\to\infty} \nu^{(k)} = \lambda_1$, and the convergence will be faster when the ratio $|\lambda_2/\lambda_1|$ is smaller.

If the matrix A is *real* and *symmetric* it can be proved, always assuming that $\alpha_1 \neq 0$, that (see [GL89], pp. 406-407)

$$|\lambda_1 - \nu^{(k)}| \leq |\lambda_1 - \lambda_n| \tan^2(\theta_0) \left|\frac{\lambda_2}{\lambda_1}\right|^{2k}, \tag{5.23}$$

where $\cos(\theta_0) = |\mathbf{x}_1^T \mathbf{q}^{(0)}| \neq 0$. Inequality (5.23) outlines that the convergence of the sequence $\nu^{(k)}$ to λ_1 is *quadratic* with respect to the ratio $|\lambda_2/\lambda_1|$ (we refer to Section 5.3.3 for numerical results).
We conclude the section by providing a stopping criterion for the iteration (5.17). For this purpose, let us introduce the residual at step k

$$\mathbf{r}^{(k)} = \mathrm{A}\mathbf{q}^{(k)} - \nu^{(k)}\mathbf{q}^{(k)}, \qquad k \geq 1,$$

and, for $\varepsilon > 0$, the matrix $\varepsilon \mathrm{E}^{(k)} = -\mathbf{r}^{(k)} \left[\mathbf{q}^{(k)}\right]^H \in \mathbb{C}^{n\times n}$ with $\|\mathrm{E}^{(k)}\|_2 = 1$. Since

$$\varepsilon \mathrm{E}^{(k)}\mathbf{q}^{(k)} = -\mathbf{r}^{(k)}, \qquad k \geq 1, \tag{5.24}$$

we obtain $\left(\mathrm{A} + \varepsilon \mathrm{E}^{(k)}\right)\mathbf{q}^{(k)} = \nu^{(k)}\mathbf{q}^{(k)}$. As a result, at each step of the power method $\nu^{(k)}$ is an *eigenvalue of the perturbed matrix* $\mathrm{A}+\varepsilon\mathrm{E}^{(k)}$. From (5.24) and from definition (1.20) it also follows that $\varepsilon = \|\mathbf{r}^{(k)}\|_2$ for $k = 1, 2, \ldots$. Plugging this identity back into (5.10) and approximating the partial derivative in (5.10) by the incremental ratio $|\lambda_1 - \nu^{(k)}|/\varepsilon$, we get

$$|\lambda_1 - \nu^{(k)}| \simeq \frac{\|\mathbf{r}^{(k)}\|_2}{|\cos(\theta_\lambda)|}, \qquad k \geq 1, \tag{5.25}$$

where θ_λ is the angle between the right and the left eigenvectors, $\mathbf{x}_1$ and $\mathbf{y}_1$, associated with λ_1. Notice that, if A is an hermitian matrix, then $\cos(\theta_\lambda) = 1$, so that (5.25) yields an estimate which is analogue to (5.13).

In practice, in order to employ the estimate (5.25) it is necessary at each step k to replace $|\cos(\theta_\lambda)|$ with the module of the scalar product between two approximations $\mathbf{q}^{(k)}$ and $\mathbf{w}^{(k)}$ of $\mathbf{x}_1$ and $\mathbf{y}_1$, computed by the power method. The following a posteriori estimate is thus obtained

$$|\lambda_1 - \nu^{(k)}| \simeq \frac{\|\mathbf{r}^{(k)}\|_2}{|(\mathbf{w}^{(k)})^H \mathbf{q}^{(k)}|}, \qquad k \geq 1. \tag{5.26}$$

Examples of applications of (5.26) will be provided in Section 5.3.3.

5.3.2 Inverse Iteration

In this section we look for an approximation of the eigenvalue of a matrix $\mathrm{A} \in \mathbb{C}^{n\times n}$ which is *closest* to a given number $\mu \in \mathbb{C}$, where $\mu \notin \sigma(\mathrm{A})$. For this, the power iteration (5.17) can be applied to the matrix $(\mathrm{M}_\mu)^{-1} = (\mathrm{A} - \mu\mathrm{I})^{-1}$, yielding the so-called *inverse iteration* or *inverse power method*. The number μ is called a *shift*.

The eigenvalues of M_μ^{-1} are $\xi_i = (\lambda_i - \mu)^{-1}$; let us assume that there exists an integer m such that

$$|\lambda_m - \mu| < |\lambda_i - \mu|, \qquad \forall i = 1, \ldots, n \qquad \text{and } i \neq m. \tag{5.27}$$

This amounts to requiring that the eigenvalue λ_m which is closest to μ has multiplicity equal to 1. Moreover, (5.27) shows that ξ_m is the eigenvalue of M_μ^{-1} with largest module; in particular, if $\mu = 0$, λ_m turns out to be the eigenvalue of A with smallest module.

Given an arbitrary initial vector $\mathbf{q}^{(0)} \in \mathbb{C}^n$ of unit Euclidean norm, for $k = 1, 2, \ldots$ the following sequence is constructed:

$$\begin{aligned} &(\mathrm{A} - \mu\mathrm{I})\,\mathbf{z}^{(k)} = \mathbf{q}^{(k-1)}, \\ &\mathbf{q}^{(k)} = \mathbf{z}^{(k)}/\|\mathbf{z}^{(k)}\|_2, \\ &\sigma^{(k)} = (\mathbf{q}^{(k)})^H \mathrm{A}\mathbf{q}^{(k)}. \end{aligned} \tag{5.28}$$

Notice that the eigenvectors of M_μ are the same as those of A since $\mathrm{M}_\mu = \mathrm{X}\,(\Lambda - \mu\mathrm{I}_n)\,\mathrm{X}^{-1}$, where $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$. For this reason, the Rayleigh quotient in (5.28) is computed directly on the matrix A (and not on M_μ^{-1}). The main difference with respect to (5.17) is that at each step k a linear system with coefficient matrix $\mathrm{M}_\mu = \mathrm{A} - \mu\mathrm{I}$ *must be solved*. For numerical convenience, the LU factorization of M_μ is computed once for all at $k = 1$, so that at each step only two triangular systems are to be solved, with a cost of the order of n^2 flops.

Although being more computationally expensive than the power method (5.17), the inverse iteration has the advantage that it can converge to any desired eigenvalue of A (namely, the one closest to the shift μ). Inverse iteration is thus ideally suited for refining an initial estimate μ of an eigenvalue of A, which can be obtained, for instance, by applying the localization techniques introduced in Section 5.1. Inverse iteration can be also effectively employed to compute the eigenvector associated with a given (approximate) eigenvalue, as described in Section 5.8.1.

In view of the convergence analysis of the iteration (5.28) we assume that A is diagonalizable, so that $\mathbf{q}^{(0)}$ can be represented in the form (5.19). Proceeding in the same way as in the power method, we let

$$\tilde{\mathbf{q}}^{(k)} = \mathbf{x}_m + \sum_{i=1, i\neq m}^{n} \frac{\alpha_i}{\alpha_m} \left(\frac{\xi_i}{\xi_m}\right)^k \mathbf{x}_i,$$

where $\mathbf{x}_i$ are the eigenvectors of M_μ^{-1} (and thus also of A), while α_i are as in (5.19). As a consequence, recalling the definition of ξ_i and using (5.27), we get

$$\lim_{k\to\infty} \tilde{\mathbf{q}}^{(k)} = \mathbf{x}_m, \qquad \lim_{k\to\infty} \sigma^{(k)} = \lambda_m.$$

Convergence will be faster when μ is closer to λ_m. Under the same assumptions made for proving (5.26), the following a posteriori estimate can be obtained for the approximation error on λ_m

$$|\lambda_m - \sigma^{(k)}| \simeq \frac{\|\widehat{\mathbf{r}}^{(k)}\|_2}{|(\widehat{\mathbf{w}}^{(k)})^H \mathbf{q}^{(k)}|}, \qquad k \geq 1, \tag{5.29}$$

where $\widehat{\mathbf{r}}^{(k)} = \mathrm{A}\mathbf{q}^{(k)} - \sigma^{(k)}\mathbf{q}^{(k)}$ and $\widehat{\mathbf{w}}^{(k)}$ is the k-th iterate of the inverse power method to approximate the left eigenvector associated with λ_m.

5.3.3 Implementation Issues

The convergence analysis of Section 5.3.1 shows that the effectiveness of the power method strongly depends on the dominant eigenvalues being *well-separated* (that is, $|\lambda_2|/|\lambda_1| \ll 1$). Let us now analyze the behavior of iteration (5.17) when *two* dominant eigenvalues of *equal* module exist (that is, $|\lambda_2| = |\lambda_1|$). Three cases must be distinguished:

1. $\lambda_2 = \lambda_1$: the two dominant eigenvalues are coincident. The method is still convergent, since for k sufficiently large (5.20) yields

$$\mathrm{A}^k\mathbf{q}^{(0)} \simeq \lambda_1^k\,(\alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2)$$

   which is an eigenvector of A. For $k \to \infty$, the sequence $\tilde{\mathbf{q}}^{(k)}$ (after a suitable redefinition) converges to a vector lying in the subspace spanned by the eigenvectors $\mathbf{x}_1$ and $\mathbf{x}_2$, while the sequence $\nu^{(k)}$ still converges to λ_1.
2. $\lambda_2 = -\lambda_1$: the two dominant eigenvalues are opposite. In this case the eigenvalue of largest module can be approximated by applying the power method to the matrix A^2. Indeed, for $i = 1, \ldots, n$, $\lambda_i(\mathrm{A}^2) = [\lambda_i(\mathrm{A})]^2$, so that $\lambda_1^2 = \lambda_2^2$ and the analysis falls into the previous case, where the matrix is now A^2.
3. $\lambda_2 = \overline{\lambda_1}$: the two dominant eigenvalues are complex conjugate. Here, undamped oscillations arise in the sequence of vectors $\mathbf{q}^{(k)}$ and the power method is not convergent (see [Wil65], Chapter 9, Section 12).

As for the computer implementation of (5.17), it is worth noting that normalizing the vector $\mathbf{q}^{(k)}$ to 1 keeps away from *overflow* (when $|\lambda_1| > 1$) or *underflow* (when $|\lambda_1| < 1$) in (5.20). We also point out that the requirement $\alpha_1 \neq 0$ (which is a priori impossible to fulfil when no information about the eigenvector $\mathbf{x}_1$ is available) is not essential for the actual convergence of the algorithm.

Indeed, although it can be proved that, working in exact arithmetic, the sequence (5.17) converges to the pair $(\lambda_2, \mathbf{x}_2)$ if $\alpha_1 = 0$ (see Exercise 10), the arising of (unavoidable) rounding errors ensures that in practice the vector $\mathbf{q}^{(k)}$ contains a *non-null* component also in the direction of $\mathbf{x}_1$. This allows for the eigenvalue λ_1 to "show-up" and the power method to quickly converge to it.
An implementation of the power method is given in Program 26. Here and in the following algorithm, the convergence check is based on the a posteriori estimate (5.26).

Here and in the remainder of the chapter, the input data `z0`, `tol` and `nmax` are the initial vector, the tolerance for the stopping test and the maximum admissible number of iterations, respectively. In output, `lambda` is the approximate eigenvalue, `relres` is the vector contain the sequence $\{\|\mathbf{r}^{(k)}\|_2/|\cos(\theta_\lambda)|\}$ (see (5.26)), whilst `x` and `iter` are the approximation of the eigenvector $\mathbf{x}_1$ and the number of iterations taken by the algorithm to converge, respectively.

Program 26 - powerm : Power method

```
function [lambda,x,iter,relres]=powerm(A,z0,tol,nmax)
%POWERM Power method
%  [LAMBDA,X,ITER,RELRES]=POWERM(A,Z0,TOL,NMAX) computes the
%  eigenvalue LAMBDA of largest module of the matrix A and the corresponding
%  eigenvector X of unit norm. TOL specifies the tolerance of the method.
%  NMAX specifies the maximum number of iterations. Z0 specifies the initial
%  guess. ITER is the iteration number at which X is computed.
q=z0/norm(z0); q2=q;
relres=tol+1; iter=0; z=A*q;
while relres(end)>=tol & iter<=nmax
 q=z/norm(z); z=A*q;
 lambda=q'*z; x=q;
 z2=q2'*A; q2=z2/norm(z2); q2=q2';
 y1=q2; costheta=abs(y1'*x);
 if costheta >= 5e-2
   iter=iter+1;
   temp=norm(z-lambda*q)/costheta;
   relres=[relres; temp];
 else
   fprintf('Multiple eigenvalue'); break;
 end
end
return
```

A coding of the inverse power method is provided in Program 27. The input parameter `mu` is the initial approximation of the eigenvalue. In output, `sigma` is the approximation of the computed eigenvalue and `relres` is a vector that contain the sequence $\left\{\|\widehat{\mathbf{r}}^{(k)}\|_2/|(\widehat{\mathbf{w}}^{(k)})^H \ \mathbf{q}^{(k)}|\right\}$ (see (5.29)). The LU factorization (with partial pivoting) of the matrix M_μ is carried out using the MATLAB intrinsic function `lu`.

Program 27 - invpower : Inverse power method

```
function [sigma,x,iter,relres]=invpower(A,z0,mu,tol,nmax)
%INVPOWER Inverse power method
%  [SIGMA,X,ITER,RELRES]=INVPOWER(A,Z0,MU,TOL,NMAX) computes the
%  eigenvalue LAMBDA of smallest module of the matrix A and the
%  corresponding eigenvector X of unit norm. TOL specifies the tolerance of the
%  method.  NMAX specifies the maximum number of iterations. X0 specifies
%  the initial guess. MU is the shift. ITER is the iteration number at which
%  X is computed.
M=A-mu*eye(size(A)); [L,U,P]=lu(M);
q=z0/norm(z0); q2=q'; sigma=[];
relres=tol+1; iter=0;
while relres(end)>=tol & iter<=nmax
    iter=iter+1;
    b=P*q;
    y=L\b; z=U\y;
    q=z/norm(z); z=A*q; sigma=q'*z;
    b=q2'; y=U'\b; w=L'\y;
    q2=w'*P; q2=q2/norm(q2); costheta=abs(q2*q);
    if costheta>=5e-2
        temp=norm(z-sigma*q)/costheta; relres=[relres,temp];
    else
        fprintf('Multiple eigenvalue'); break;
    end
    x=q;
end
return
```

Example 5.3 Let us consider the following matrices

$$\mathrm{A} = \begin{bmatrix} 15 & -2 & 2 \\ 1 & 10 & -3 \\ -2 & 1 & 0 \end{bmatrix}, \quad \mathrm{V} = \begin{bmatrix} -0.944 & 0.393 & -0.088 \\ -0.312 & 0.919 & 0.309 \\ 0.112 & 0.013 & 0.947 \end{bmatrix}. \tag{5.30}$$

Matrix A has the following eigenvalues (to five significant figures): $\lambda_1 = 14.103$, $\lambda_2 = 10.385$ and $\lambda_3 = 0.512$, while the corresponding eigenvectors are the vector columns of matrix V.

To approximate the pair $(\lambda_1, \mathbf{x}_1)$, we have run the Program 26 with initial datum $\mathbf{z}^{(0)} = [1, 1, 1]^T$. After 71 iterations of the power method the absolute errors are $|\lambda_1 - \nu^{(71)}| = 2.2341 \cdot 10^{-10}$ and $\|\mathbf{x}_1 - \mathbf{x}_1^{(71)}\|_\infty = 1.42 \cdot 10^{-11}$.

In a second run, we have used $\mathbf{z}^{(0)} = \mathbf{x}_2 + \mathbf{x}_3$ (notice that with this choice $\alpha_1 = 0$). After 215 iterations the absolute errors are $|\lambda_1 - \nu^{(215)}| = 4.26 \cdot 10^{-14}$ and $\|\mathbf{x}_1 - \mathbf{x}_1^{(215)}\|_\infty = 1.38 \cdot 10^{-14}$.

Figure 5.2 (*left*) shows the reliability of the a posteriori estimate (5.26). The sequences $|\lambda_1 - \nu^{(k)}|$ (*solid line*) and the corresponding a posteriori estimates (5.26)

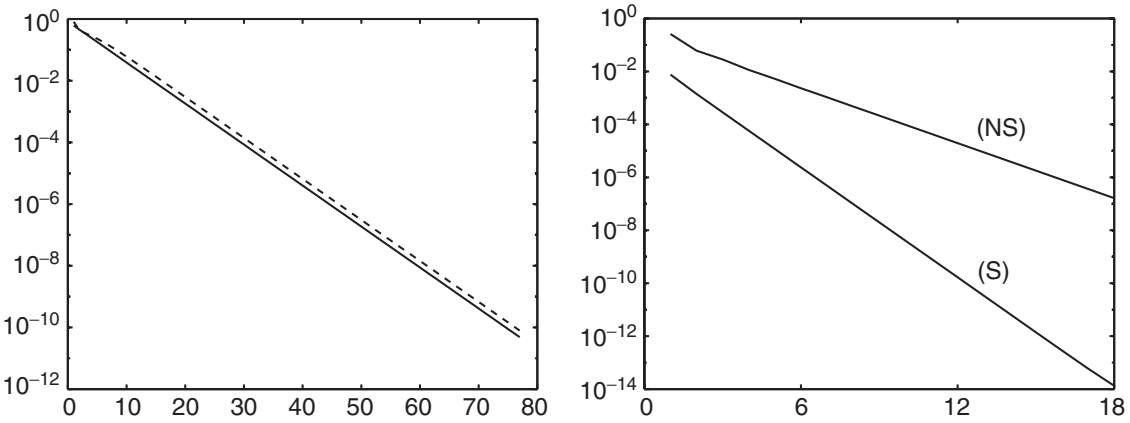

Fig. 5.2. Comparison between the a posteriori error estimate and the actual absolute error for matrix A in (5.30) (*left*); convergence curves for the power method applied to matrix A in (5.31) in its symmetric (S) and nonsymmetric (NS) forms (*right*)

(*dashed line*) are plotted as a function of the number of iterations (in abscissae). Notice the excellent agreement between the two curves.
Let us now consider the matrices

$$\mathrm{A} = \begin{bmatrix} 1 & 3 & 4 \\ 3 & 1 & 2 \\ 4 & 2 & 1 \end{bmatrix}, \ \mathrm{T} = \begin{bmatrix} 8 & 1 & 6 \\ 3 & 5 & 7 \\ 4 & 9 & 2 \end{bmatrix} \tag{5.31}$$

where A has the following spectrum: $\lambda_1 = 7.047$, $\lambda_2 = -3.1879$ and $\lambda_3 = -0.8868$ (to five significant figures).
It is interesting to compare the behaviour of the power method when computing λ_1 for the symmetric matrix A and for its similar matrix $\mathrm{M} = \mathrm{T}^{-1}\mathrm{AT}$, where T is the nonsingular (and nonorthogonal) matrix in (5.31).
Running Program 26 with $\mathbf{z}^{(0)} = [1, 1, 1]^T$, the power method converges to the eigenvalue λ_1 in 18 and 30 iterations, for matrices A and M, respectively. The sequence of absolute errors $|\lambda_1 - \nu^{(k)}|$ is plotted in Figure 5.2 (right) where (S) and (NS) refer to the computations on A and M, respectively. Notice the rapid error reduction in the symmetric case, according to the quadratic convergence properties of the power method (see Section 5.3.1).
We finally employ the inverse power method (5.28) to compute the eigenvalue of smallest module $\lambda_3 = 0.512$ of matrix A in (5.30). Running Program 27 with $\mathbf{q}^{(0)} = [1, 1, 1]^T/\sqrt{3}$, the method converges in 9 iterations, with absolute errors $|\lambda_3 - \sigma^{(9)}| = 1.194 \cdot 10^{-12}$ and $\|\mathbf{x}_3 - \mathbf{x}_3^{(9)}\|_\infty = 4.59 \cdot 10^{-13}$. •

5.4 The QR Iteration

In this section we present some iterative techniques for *simultaneously* approximating *all* the eigenvalues of a given matrix A. The basic idea consists of reducing A, by means of suitable similarity transformations, into a form for which the calculation of the eigenvalues is easier than on the starting matrix.

The problem would be satisfactorily solved if the unitary matrix U of the Schur decomposition theorem 1.5, such that $\mathrm{T} = \mathrm{U}^H \mathrm{A}\mathrm{U}$, T being upper triangular and with $t_{ii} = \lambda_i(\mathrm{A})$ for $i = 1, \ldots, n$, could be determined in a direct way, that is, with a finite number of operations. Unfortunately, it is a consequence of Abel's theorem that, for $n \geq 5$, the matrix U cannot be computed in an elementary way (see Exercise 8). Thus, our problem can be solved only resorting to iterative techniques.
The reference algorithm in this context is the *QR iteration* method, that is here examined only in the case of real matrices. (For some remarks on the extension of the algorithms to the complex case, see [GL89], Section 5.2.10 and [Dem97], Section 4.2.1).

Let $\mathrm{A} \in \mathbb{R}^{n \times n}$; given an orthogonal matrix $\mathrm{Q}^{(0)} \in \mathbb{R}^{n \times n}$ and letting $\mathrm{T}^{(0)} = (\mathrm{Q}^{(0)})^T \mathrm{A}\mathrm{Q}^{(0)}$, for $k = 1, 2, \ldots$, until convergence, the QR iteration consists of:

$$\begin{aligned} &\text{determine } \mathrm{Q}^{(k)}, \mathrm{R}^{(k)} \text{ such that} \\ &\mathrm{Q}^{(k)}\mathrm{R}^{(k)} = \mathrm{T}^{(k-1)} \qquad \text{(QR factorization);} \\ &\text{then, let} \\ &\mathrm{T}^{(k)} = \mathrm{R}^{(k)}\mathrm{Q}^{(k)}. \end{aligned} \tag{5.32}$$

At each step $k \geq 1$, the first phase of the iteration is the factorization of the matrix $\mathrm{T}^{(k-1)}$ into the product of an orthogonal matrix $\mathrm{Q}^{(k)}$ with an upper triangular matrix $\mathrm{R}^{(k)}$ (see Section 5.6.3). The second phase is a simple matrix product. Notice that

$$\begin{aligned} \mathrm{T}^{(k)} &= \mathrm{R}^{(k)}\mathrm{Q}^{(k)} = (\mathrm{Q}^{(k)})^T(\mathrm{Q}^{(k)}\mathrm{R}^{(k)})\mathrm{Q}^{(k)} = (\mathrm{Q}^{(k)})^T\mathrm{T}^{(k-1)}\mathrm{Q}^{(k)} \\ &= (\mathrm{Q}^{(0)}\mathrm{Q}^{(1)}\cdots\mathrm{Q}^{(k)})^T\mathrm{A}(\mathrm{Q}^{(0)}\mathrm{Q}^{(1)}\cdots\mathrm{Q}^{(k)}), \qquad k \geq 0, \end{aligned} \tag{5.33}$$

i.e., every matrix $\mathrm{T}^{(k)}$ is *orthogonally similar* to A. This is particularly relevant for the *stability* of the method, since, as shown in Section 5.2, the conditioning of the matrix eigenvalue problem for $\mathrm{T}^{(k)}$ is not worse than it is for A (see also [GL89], p. 360).

A basic implementation of the QR iteration (5.32), assuming $\mathrm{Q}^{(0)} = \mathrm{I}_n$, is examined in Section 5.5, while a more computationally efficient version, starting from $\mathrm{T}^{(0)}$ in upper Hessenberg form, is described in detail in Section 5.6. If A has real eigenvalues, distinct in module, it will be seen in Section 5.5 that the limit of $\mathrm{T}^{(k)}$ is an upper triangular matrix (with the eigenvalues of A on the main diagonal). However, if A has complex eigenvalues the limit of $\mathrm{T}^{(k)}$ *cannot* be an upper triangular matrix T. Indeed if it were T would necessarily have real eigenvalues, although it is similar to A.

Failure to converge to a triangular matrix may also happen in more general situations, as addressed in Example 5.9.

For this, it is necessary to introduce variants of the QR iteration (5.32), based on deflation and *shift* techniques (see Section 5.7 and, for a more detailed

discussion of the subject, [GL89], Chapter 7, [Dat95], Chapter 8 and [Dem97], Chapter 4).
These techniques allow for $\mathrm{T}^{(k)}$ to converge to an upper *quasi-triangular* matrix, known as the *real Schur decomposition* of A, for which the following result holds (for the proof we refer to [GL89], pp. 341-342).

Property 5.8 *Given a matrix* $\mathrm{A} \in \mathbb{R}^{n\times n}$*, there exists an orthogonal matrix* $\mathrm{Q} \in \mathbb{R}^{n\times n}$ *such that*

$$\mathrm{Q}^T\mathrm{AQ} = \begin{bmatrix} \mathrm{R}_{11} & \mathrm{R}_{12} & \dots & \mathrm{R}_{1m} \\ 0 & \mathrm{R}_{22} & \dots & \mathrm{R}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathrm{R}_{mm} \end{bmatrix}, \tag{5.34}$$

where each block R_{ii} *is either a real number or a matrix of order 2 having complex conjugate eigenvalues, and*

$$\mathrm{Q} = \lim_{k\to\infty} \left[\mathrm{Q}^{(0)}\mathrm{Q}^{(1)}\cdots\mathrm{Q}^{(k)}\right] \tag{5.35}$$

$\mathrm{Q}^{(k)}$ *being the orthogonal matrix generated by the k-th factorization step of the QR iteration* (5.32)*.*

The QR iteration can be also employed to compute all the eigenvectors of a given matrix. For this purpose, we describe in Section 5.8 two possible approaches, one based on the coupling between (5.32) and the inverse iteration (5.28), the other working on the real Schur form (5.34).

5.5 The Basic QR Iteration

In the basic version of the QR method, one sets $\mathrm{Q}^{(0)} = \mathrm{I}_n$ in such a way that $\mathrm{T}^{(0)} = \mathrm{A}$. At each step $k \geq 1$ the QR factorization of the matrix $\mathrm{T}^{(k-1)}$ can be carried out using the modified Gram-Schmidt procedure introduced in Section 3.4.3, with a cost of the order of $2n^3$ flops (for a full matrix A). The following convergence result holds (for the proof, see [GL89], Theorem 7.3.1, or [Wil65], pp. 517-519).

Property 5.9 (Convergence of QR method) *Let* $\mathrm{A} \in \mathbb{R}^{n\times n}$ *be a matrix with real eigenvalues such that*

$$|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n|.$$

Then

$$\lim_{k\to+\infty} \mathrm{T}^{(k)} = \begin{bmatrix} \lambda_1 & t_{12} & \dots & t_{1n} \\ 0 & \lambda_2 & t_{23} & \dots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n \end{bmatrix}. \tag{5.36}$$

As for the convergence rate, we have

$$|t_{i,i-1}^{(k)}| = \mathcal{O}\left(\left|\frac{\lambda_i}{\lambda_{i-1}}\right|^k\right), \qquad i = 2, \dots, n, \qquad \text{for } k \to +\infty. \tag{5.37}$$

Under the additional assumption that A *is symmetric, the sequence* $\{\mathrm{T}^{(k)}\}$ *tends to a diagonal matrix.*

If the eigenvalues of A, although being distinct, are *not well-separated*, it follows from (5.37) that the convergence of $\mathrm{T}^{(k)}$ towards a triangular matrix can be quite slow. With the aim of accelerating it, one can resort to the so-called *shift* technique, which will be addressed in Section 5.7.

Remark 5.2 It is always possible to reduce the matrix A into a triangular form by means of an iterative algorithm employing *nonorthogonal* similarity transformations. In such a case, the so-called *LR iteration* (known also as *Rutishauser method*, [Rut58]) can be used, from which the QR method has actually been derived (see also [Fra61], [Wil65]). The LR iteration is based on the factorization of the matrix A into the product of two matrices L and R, respectively unit lower triangular and upper triangular, and on the (nonorthogonal) similarity transformation

$$\mathrm{L}^{-1}\mathrm{AL} = \mathrm{L}^{-1}(\mathrm{LR})\mathrm{L} = \mathrm{RL}.$$

The rare use of the LR method in practical computations is due to the loss of accuracy that can arise in the LR factorization because of the increase in module of the upper diagonal entries of R. This aspect, together with the details of the implementation of the algorithm and some comparisons with the QR method, is examined in [Wil65], Chapter 8. ■

Example 5.4 We apply the QR method to the symmetric matrix $\mathrm{A} \in \mathbb{R}^{4\times 4}$ such that $a_{ii} = 4$, for $i = 1, \dots, 4$, and $a_{ij} = 4 + i - j$ for $i < j \leq 4$, whose eigenvalues are (to three significant figures) $\lambda_1 = 11.09$, $\lambda_2 = 3.41$, $\lambda_3 = 0.90$ and $\lambda_4 = 0.59$. After 20 iterations, we get

$$\mathrm{T}^{(20)} = \begin{bmatrix} \boxed{11.09} & 6.44 \cdot 10^{-10} & -3.62 \cdot 10^{-15} & 9.49 \cdot 10^{-15} \\ 6.47 \cdot 10^{-10} & \boxed{3.41} & 1.43 \cdot 10^{-11} & 4.60 \cdot 10^{-16} \\ 1.74 \cdot 10^{-21} & 1.43 \cdot 10^{-11} & \boxed{0.90} & 1.16 \cdot 10^{-4} \\ 2.32 \cdot 10^{-25} & 2.68 \cdot 10^{-15} & 1.16 \cdot 10^{-4} & \boxed{0.58} \end{bmatrix}.$$