Numerical linear algebra autumn 2004

Problem set 8

1. What happens if you apply unshifted QR–algorithm to an orthogonal matrix?

2. Consider the following matrix

$$A = \begin{pmatrix} 2 & -1 \\ 1 & 2 \end{pmatrix}$$

   Check that the QR–algorithm cannot converge with real shifts. What happens if you use complex shifts?

3. In the unshifted QR–algorithm we had $A^k = \underline{Q_k}\,\underline{R_k}$. If μ_k are the shifts show that in the shifted case we have

$$(A - \mu_k I)...(A - \mu_1 I) = \underline{Q_k}\,\underline{R_k}$$

4. Take matrices of different size and compare the times required for matrix multiplication and computation of eigenvalues with `eig` . `eig` computes first the Hessenberg/tridiagonal form of the matrix. So given a matrix, the actual time required by the QR–algorithm itself is essentially the time required by `eig` minus the time required by `hess` . Check in particular that computing the eigenvalues of the tridiagonal matrix is very fast.

5. When manipulating big and sparse matrices it is best not to store the zeros. This kind of data structure is implemented in Matlab with command `sparse`. Give an algorithm for computing matrix vector product with this data structure.

Homework

Do a program which computes QR–decomposition of a Hessenberg matrix using either Givens rotations or 2×2 Householder reflectors. Check that this is much faster than QR–decomposition of a general matrix.